

С. К. Гурский
С. В. Домников
О. И. Александров

ОПТИМИЗАЦИЯ РЕЖИМОВ РАБОТЫ ЭНЕРГОСИСТЕМ

ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Допущено Министерством высшего и среднего
специального образования БССР в качестве
учебного пособия для энергетических
специальностей вузов

ББК 31.27я73

Г95

УДК 621.311.004.13 (076.5)

Рецензенты: кафедра электрических систем Ивановского энергетического института; Ю.В.Щербина, д-р техн. наук, проф. кафедры автоматизации энергосистем Киевского политехнического института

Гурский С.К. и др.

Г95 Оптимизация режимов работы энергосистем: Лаб. практикум. [Для энерг. спец. вузов] / С.К.Гурский, С.В.Домников, О.И.Александров. — Мн.: Выш. шк., 1985. — 147 с., ил.

40 к.

Содержит детальное описание комплекса лабораторных работ по одноименному курсу и позволяет студентам практически овладеть такими важнейшими разделами курса, как наиболее часто используемые в электроэнергетике математические методы оптимизации; оптимальное распределение нагрузок между электростанциями и их агрегатами с выбором состава последних; решение задач АСДУ энергосистемами под управлением операционной системы ДОС/ЕС; использование пакетов прикладных программ в составе АСДУ энергосистемами. В приложении приведен текст исходных модулей программ для ЕС ЭВМ на языке ФОРТРАН IV.

ББК 31.27я73

2302040000—106
Г 35 85
М304 (05) -85

© Издательство "Высшая школа", 1985.

ПРЕДИСЛОВИЕ

По мере роста масштабов производства и повышения производительности труда в народном хозяйстве все более актуальными становятся проблемы оптимального использования энергоресурсов, надежного и качественного энергоснабжения потребителей на базе Единой энергетической системы (ЕЭС) СССР. Последняя представляет собой управляемый объект, не имеющий аналогов по сложности своей структуры и многообразию производственно-технологических и организационно-экономических процессов. Обеспечение требуемого качества управления таким объектом невозможно без самого широкого внедрения в практику научных методов управления на основе новейшей вычислительной техники и других современных технических средств, что обусловило, начиная с 70-х годов, проведение работ по созданию и совершенствованию отраслевой автоматизированной системы управления (ОАСУ) "Энергия" в качестве одной из актуальнейших народнохозяйственных задач. Важнейшей и специфической для энергетики подсистемой является автоматизированная система диспетчерского управления (АСДУ) СССР, предназначенная для оптимального планирования режимов работы энергосистем и непосредственного технологического управления ими.

Работа в качестве пользователя, а тем более разработчика оптимизационных задач требует хорошего знания возможностей общего и специального математического обеспечения вычислительных машин третьего поколения — ЕС ЭВМ (операционных систем, банков данных, пакетов прикладных программ) и достаточно развитых навыков по их эксплуатации в условиях управления производственными системами. Соответствующая подготовка студентов выходит за рамки программы и возможностей курса "Программирование и применение ЭВМ" и должна приобретаться в процессе выполнения лабораторного практикума по курсу "Оптимизация режимов работы энергосистем" наряду с изучением методов оптимизации в соответствующих лекционных курсах.

В настоящее время отсутствуют какие-либо учебные пособия или методические рекомендации по организации данного лабораторного практикума, несмотря на потребность в нем вузов страны, готовящих инженеров по специальностям 0302, 0304, 0301 и частично 0303, 0314.

Настоящее учебное пособие написано применительно для операционной системы ДОС/ЕС (версия — 2.1). В основу разработанного авторами лабораторного практикума положены следующие принципы.

1. Состав практикума и содержание лабораторных работ позволяют студентам закрепить усвоение основных математических методов оптимизации, используемых в оптимизационных задачах (метод неопределенных множителей Лагранжа, метод динамического программирования); ознакомиться на практике с характерными примерами задач оптимизации режимов энергосистем; изучить возможности операционной системы ДОС/ЕС в объеме, необходимом для инженера-пользователя; получить первичные навыки работы с базами данных и пакетами прикладных программ для решения конкретных производственных задач.

2. Используемые при выполнении лабораторных работ программы для ЕС ЭВМ в совокупности образуют единый комплекс для решения одной задачи АСДУ с целью развития у студентов навыков организации взаимодействия различных машинных программ и файлов данных в разнообразных ситуациях, возникающих в процессе эксплуатации и управления энергосистемами.

3. Постановка подзадач оптимизации режимов энергосистем, принятая в лабораторном практикуме, минимально упрощена и носит по существу учебный характер с целью экономии времени, затрачиваемого студентами на подготовку исходных данных, и сокращения объема машинного времени, выделяемого ВЦ на проведение лабораторных работ по курсу.

4. Ввиду объемности работ и комплексного характера практикума предполагается, что выполнению каждой работы студентами должна предшествовать внеаудиторная предварительная подготовка.

Инструкции к лабораторным работам включают краткие теоретические сведения, описание программы решения задачи на ЕС ЭВМ, задания на предварительную подготовку к выполнению лабораторных работ, порядок выполнения работ в лаборатории, требуемое содержание отчета по каждой работе, контрольные вопросы и варианты исходных данных.

Порядок выполнения каждой работы в лаборатории единый и содержит следующие пункты: 1) изучить инструкцию к данной лабораторной работе; 2) предъявить преподавателю подготовленную исходную информацию по заданному варианту для проверки правильности кодирования; 3) получить у преподавателя соответствующую колоду перфокарт заданий для ДЭС/ЕС; 4) в порядке очереди, установленном преподавателем, передать колоду перфокарт оператору ЕС ЭВМ для установки во входной поток заданий; 5) получить у оператора распечатку исходных данных и сообщения операционной системы на системном устройстве печати о результатах обработки задания.

Авторы выражают свою глубокую благодарность профессору Ю.В.Щербине и доцентам Б.Я.Прахину и В.А.Баженову за полезные советы и замечания, которые помогли авторам в работе над рукописью на заключительном этапе.

Авторы заранее выражают благодарность всем, кто сочтет необходимым сообщить о недостатках и погрешностях данного пособия, и просят высылать все рекомендации и замечания по адресу: 220048, Минск, проспект Машерова, 11, издательство "Высшая школа".

Авторы

ЛАБОРАТОРНАЯ РАБОТА 1

ПРИМЕНЕНИЕ МЕТОДА ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ К ОПТИМИЗАЦИИ РАСПРЕДЕЛЕНИЯ АКТИВНЫХ МОЩНОСТЕЙ МЕЖДУ АГРЕГАТАМИ ЭЛЕКТРОСТАНЦИИ

Ц е л ь р а б о т ы : ознакомиться с реализацией метода динамического программирования на примере простейшей постановки задачи распределения активных мощностей между агрегатами электростанций, уяснить условия применимости этого метода и метода неопределенных множителей Лагранжа, получить навыки в решении практических энергетических задач с помощью имеющихся программ для ЕС ЭВМ.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В простейшей постановке задача оптимального распределения активных мощностей между генераторами электростанции решается отдельно для каждого часа суток и преследует цель найти такие величины активных мощностей генераторов, при которых заданная суммарная активная мощность станции в течение рассматриваемого часа будет вырабатываться с минимальным расходом топлива. Если не учитывать потребления активной мощности на собственные нужды электростанции, то указанную задачу при блочной схеме станции (каждый блок или агрегат включает в себя котел, паровую турбину и генератор) можно сформулировать следующим образом:

$$B = \sum_{\tau=1}^n B_{\tau} = \sum_{\tau=1}^n f_{\tau}(P_{\tau}) \rightarrow \min; \quad (1)$$

$$\sum_{\tau=1}^n P_{\tau} = P; \quad (2)$$

$$P_{\tau \min} \leq P_{\tau} \leq P_{\tau \max}, \quad \tau = 1, 2, \dots, n, \quad (3)$$

где P , B — соответственно заданная активная мощность станции и суммарный расход топлива на ней; P_{τ} — искомая активная мощность агрегата с номером τ ; $P_{\tau \min}$, $P_{\tau \max}$ — соответственно технический минимум нагрузки и располагаемая мощность агрегата; $f_{\tau}(P_{\tau})$ — расходная характеристика, т.е. зависимость расхода топлива агрегата от мощности, считающейся неизменной в течение часа; n — количество агрегатов на станции.

Выражение (1) называется *целевой функцией* или *критерием оптимальности*, а выражения (2), (3) — *ограничениями задачи оптимизации*.

До последнего времени наиболее употребительным методом решения задачи (1)–(3) был метод удельных приростов, являющийся следствием общего математического способа решения задач оптимизации с ограничениями в виде равенств (метода неопределенных множителей Лагранжа), из которого вытекает, в частности, следующий факт. Если расходные характеристики

агрегатов $f_{\tau}(P_{\tau})$ являются дифференцируемыми функциями и некоторый набор мощностей $P_{\tau}, \tau = 1, 2, \dots, n$, удовлетворяющих уравнению (2) и условиям (3) при строгих неравенствах, обеспечивает минимум суммарного расхода топлива B , то при этих значениях P_{τ} будет выполнено условие

$$\frac{dB_1}{dP_1} = \frac{dB_2}{dP_2} = \dots = \frac{dB_n}{dP_n}, \quad (4)$$

называемое обычно *правилом равенства удельных приростов*. Метод удельных приростов заключается в отыскании таких величин $P_{\tau}, \tau = 1, 2, \dots, n$, при которых выполняются условие (4) и условия (2), (3).

В тех случаях, когда набора значений $P_{\tau}, \tau = 1, 2, \dots, n$, обеспечивающего выполнение всех указанных условий, не существует, метод удельных приростов также может быть применен, если принять допущение о неоднозначности значений dB_{τ}/dP_{τ} в граничных точках интервалов (3).

Рекомендуется следующий алгоритм для определения указанных значений P_{τ} .

1. Для каждого агрегата $\tau = 1, 2, \dots, n$ по заданной расходной характеристике $B_{\tau} = f_{\tau}(P_{\tau})$ определяются величины

$$\varepsilon_{\tau} = \frac{dB_{\tau}}{dP_{\tau}}$$

при различных значениях мощности P в диапазоне $P_{\tau \min} \leq P_{\tau} \leq P_{\tau \max}$ и строятся зависимости $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau})$ (характеристики удельных приростов) на одном графике (рис. 1).

2. По рис. 1 определяются величины

$$\varepsilon_{\min} = \min_{\tau=1,2,\dots,n} \varepsilon_{\tau \min}; \quad \varepsilon_{\max} = \max_{\tau=1,2,\dots,n} \varepsilon_{\tau \max},$$

и характеристики удельных приростов (ХУП) $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau}), \tau = 1, 2, \dots, n$, дополняются вертикальными участками в точках с координатами $(P_{\tau \min}, \varepsilon_{\tau \min})$,

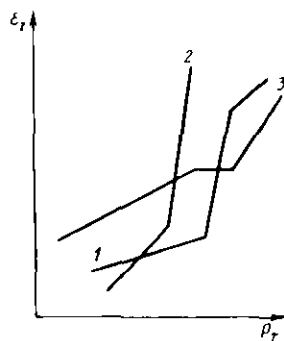


Рис. 1.

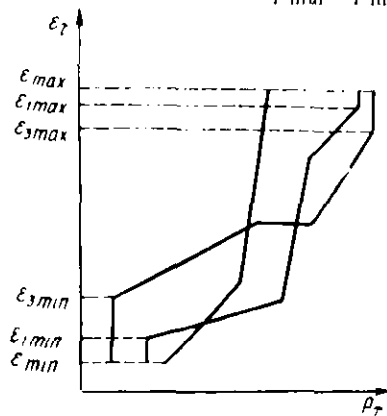


Рис. 2.

$(P_{\tau \max}, \varepsilon_{\tau \max})$ таким образом, чтобы для всех $\tau = 1, 2, \dots, n$ оказались выполненными условия (рис. 2):

$$\varepsilon_{\tau \min} = \varepsilon_{\min}; \quad \varepsilon_{\tau \max} = \varepsilon_{\max}.$$

3. Строится ХУП электростанции $\varepsilon = \varphi(P)$ так, чтобы координаты (P, ε) каждой ее точки удовлетворяли условиям:

$$P = \sum_{\tau=1}^n P_{\tau};$$

$$\varepsilon = \varepsilon_{\tau}, \quad \tau = 1, 2, \dots, n,$$

где $(P_{\tau}, \varepsilon_{\tau})$ — координаты некоторой точки характеристики $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau})$.

Это построение легко произвести на графике, для чего выполняются следующие действия.

3.1) задается набор равноотстоящих значений ординат

$$\varepsilon^{(1)}, \varepsilon^{(2)}, \dots, \varepsilon^{(\nu)}, \dots, \varepsilon^{(N)},$$

где

$$\varepsilon^{(1)} = \varepsilon_{\min}; \quad \varepsilon^{(N)} = \varepsilon_{\max}; \quad \varepsilon^{(\nu+1)} = \varepsilon^{(\nu)} + \Delta\varepsilon, \quad \nu = 1, 2, \dots, N-1;$$

$\Delta\varepsilon > 0$ — произвольно выбранное достаточно малое значение шага изменения ε ;

3.2) для каждой ординаты $\varepsilon^{(\nu)}$, $\nu = 1, 2, \dots, N$, определяется соответствующая абсцисса характеристики $\varepsilon = \varphi(P)$, для чего с помощью характеристик $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau})$, $\tau = 1, 2, \dots, n$, находятся значения мощностей $P_{\tau}^{(\nu)}$, соответствующие величинам $\varepsilon_{\tau} = \varepsilon^{(\nu)}$:

$$P_{\tau}^{(\nu)} = \varphi_{\tau}^{-1}(\varepsilon^{(\nu)}),$$

где φ_{τ}^{-1} — символ обратной функции по отношению к функции $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau})$ (рис. 3), и вычисляется искомое значение

$$P^{(\nu)} = \sum_{\tau=1}^n P_{\tau}^{(\nu)};$$

3.3) по полученному набору точек

$$(P^{(1)}, \varepsilon^{(1)}), (P^{(2)}, \varepsilon^{(2)}), \dots, (P^{(\nu)}, \varepsilon^{(\nu)}), \dots, (P^{(N)}, \varepsilon^{(N)})$$

характеристика $\varepsilon = \varphi(P)$ строится на том же графике, что и кривые $\varepsilon_{\tau} = \varphi_{\tau}(P_{\tau})$, $\tau = 1, 2, \dots, n$ (рис. 4).

4. По заданной суммарной мощности станции P с помощью характеристики $\varepsilon = \varphi(P)$ находится соответствующее значение ε и определяются искомые значения P_{τ} (см. рис. 4):

$$P_{\tau} = \varphi_{\tau}^{-1}(\varepsilon), \quad \tau = 1, 2, \dots, n.$$

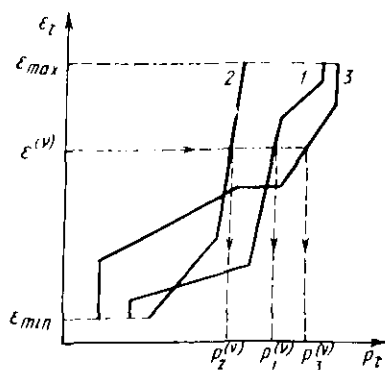


Рис. 3.

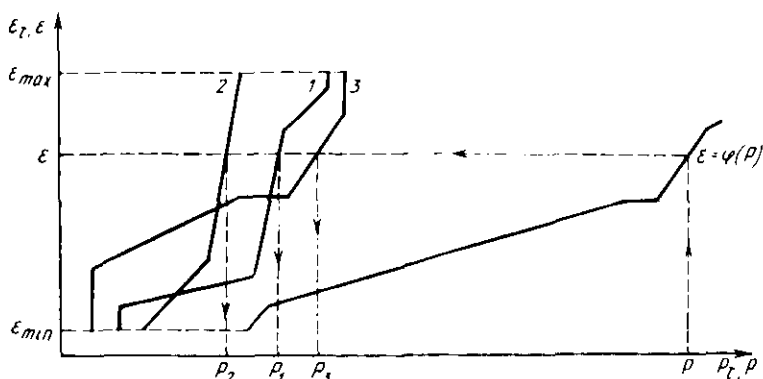


Рис. 4.

Благодаря простоте описанного алгоритма задача легко решается вручную, что и обусловило широкое использование метода в тот период, когда еще не были созданы АСУ энергосистемами, функционирующие на базе цифровых ЭВМ. Однако даже при простейшей постановке задачи оптимизации (1) – (3) метод удельных приростов не всегда применим, так как правило равенства удельных приростов (4) и уравнение (2) представляют собой необходимое, но недостаточное условие минимума целевой функции (1). Чтобы при найденных по описанному алгоритму мощностях агрегатов P_τ , $\tau = 1, 2, \dots, n$ действительно достигался минимум расхода топлива на станции, условия (2), (4) должны быть дополнены требованием выпуклости расходных характеристик агрегатов $B_\tau = f_\tau(P_\tau)$.

С учетом известного признака выпуклости дважды дифференцируемой функции это означает, что должны быть выполнены неравенства

$$\frac{d^2 B_\tau}{dP_\tau^2} > 0, \quad \tau = 1, 2, \dots, n, \quad (5)$$

во всех точках интервалов (3). На рис. 5 расходная характеристика 1 является выпуклой, а характеристика 2 — невыпуклой.

Применительно к ХУП $\varepsilon_\tau = \varphi_\tau(P_\tau)$ неравенства (5) можно записать в виде

$$\frac{d\varepsilon}{dP_\tau} \geq 0, \quad \tau = 1, 2, \dots, n,$$

и это означает, что правило равенства удельных приростов применимо только при неубывающих ХУП агрегатов. В этом легко убедиться наглядно с помощью рис. 6.

На рис. 6, а ХУП двух параллельных работающих агрегатов являются неубывающими. Пусть в исходном режиме:

$$P_1 + P_2 = P; \quad \varepsilon_1 = \varepsilon_2, \quad (6)$$

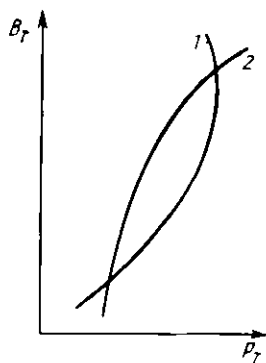


Рис. 5.

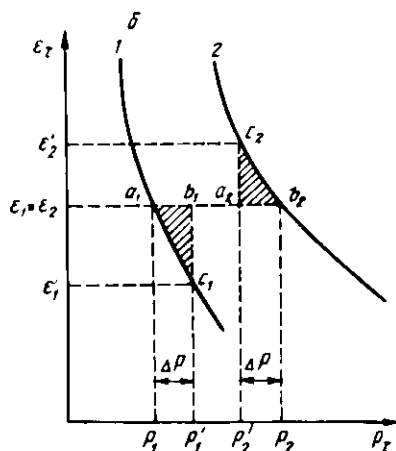
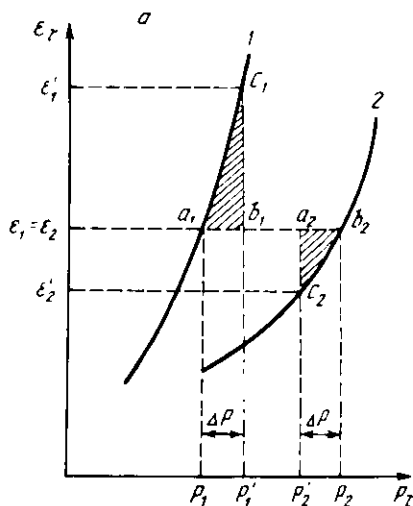


Рис. 6.

Если произвести перераспределение нагрузок между агрегатами, определяя новые нагрузки как

$$P_1' = P_1 + \Delta P; \quad P_2' = P_2 - \Delta P$$

(нагрузки P_1', P_2' должны по-прежнему удовлетворять уравнению (6)), то расходы топлива на агрегатах изменятся на величины

$$\Delta B_1 = \int_{P_1}^{P_1'} \varepsilon_1 dP_1 = F_{P_1 a_1 c_1 b_1 P_1'};$$

$$\Delta B_2 = \int_{P_2}^{P_2'} \varepsilon_2 dP_2 = - \int_{P_2'}^{P_2} \varepsilon_2 dP_2 = -F_{P_2' c_2 b_2 P_2},$$

где $F_{P_1 a_1 c_1 b_1 P_1'}$; $F_{P_2' c_2 b_2 P_2}$ — площади криволинейных трапеций на рис. 6, а.

Расход топлива на станции изменится на величину

$$\Delta B = \Delta B_1 + \Delta B_2 = F_{P_1 a_1 c_1 b_1 P_1'} - F_{P_2' c_2 b_2 P_2} = (F_{P_1 a_1 b_1 P_1'} + F_{a_1 c_1 b_1}) - (F_{P_2' a_2 b_2 P_2} - F_{c_2 a_2 b_2}) = F_{a_1 c_1 b_1} + F_{c_2 a_2 b_2} > 0.$$

Следовательно, отступление от правила равенства удельных приростов привело к перерасходу топлива на величину суммы площадей заштрихованных треугольников. Наоборот, из рис. 6, б видно, что аналогичное перераспределение нагрузок в случае убывающих ХУП дает экономию топлива, так как

$$\Delta B = -F_{a_1 b_1 c_1} - F_{a_2 b_2 c_2} < 0,$$

а это означает, что исходный режим, при котором $\varepsilon_1 = \varepsilon_2$, не оптимален.

Так как в действительности ХУП агрегатов не во всех случаях являются неубывающими во всем диапазоне нагрузок, то метод удельных приростов не всегда может быть использован на практике. Недостаток этого метода состоит также в следующем: режиму может соответствовать не глобальный, а локальный минимум расхода топлива, даже если режим находится в области ХУП, в которой удельные приросты являются неубывающими. Это означает, что отступление от указанного режима любым способом приводит к перерасходу топлива, если это отступление достаточно малое, но если режим изменить в значительной степени, то можно достичь и экономии топлива.

Указанных недостатков лишен метод динамического программирования (МДП), который позволяет найти глобальный минимум целевой функции (1) независимо от вида ХУП агрегатов.

Основные положения, на которых базируется решение задачи (1)–(3) на основе МДП, состоят в следующем. Процесс решения задачи складывается из двух этапов — прямого и обратного ходов.

Прямой ход заключается в построении двух функций двух переменных — функции Беллмана $B(Y, \tau)$ и зависимости $F(Y, \tau) = P_{\tau}^{\text{опт}}(Y)$, которые определяются следующим образом.

При $\tau = 1$

$$B(Y, 1) = f_1(Y),$$

$$P_1^{\text{опт}}(Y) = Y,$$

где переменная Y изменяется в интервале

$$P_{1 \min} \leq Y \leq P_{1 \max};$$

$f_1(Y)$ — расходная характеристика первого агрегата.

При любом фиксированном τ , $1 < \tau \leq n$.

$$B(Y, \tau) = \min [f_{\tau}(P_{\tau}) + B(Y - P_{\tau}, \tau - 1)], \quad (7)$$

а значения функции $P_{\tau}^{\text{опт}}(Y)$ определяются как значения P_{τ} , при которых достигается минимум выражения, записанного в квадратных скобках в формуле (7), для соответствующих значений независимой переменной Y .

Физический смысл независимой переменной Y и значений функции $B(Y, \tau)$ следующий. Переменная Y представляет собой суммарную нагрузку на агрегаты с номерами $i = 1, 2, \dots, \tau$ (прямой ход МДП выполняется независимо от заданного значения суммарной нагрузки станции P), которая изменяется в интервале

$$\sum_{i=1}^{\tau} P_{i \min} \leq Y \leq \sum_{i=1}^{\tau} P_{i \max}.$$

Значение функции $B(Y, \tau)$ для данного значения Y равно суммарному расходу топлива на агрегатах с номерами $i = 1, 2, \dots, \tau$ при условии, что суммарная нагрузка Y распределяется между этими агрегатами оптимально.

В соответствии с выражением (7), которое представляет собой известный в теории оптимизации принцип оптимальности Беллмана, оптимальное распределение нагрузки Y между τ агрегатами сводится к решению задачи оптимизации по одной переменной P_{τ} при условии, что к этому моменту уже известны значения функции Беллмана $B(Y, \tau-1)$. В сведении исходной задачи оптимизации по многим переменным к многократному решению задач оптимизации по одной переменной и заключается основная идея МДП. При оптимизации по P_{τ} при фиксированном Y переменная P_{τ} должна изменяться в интервале $P_{\tau \min}(Y) \leq P_{\tau} \leq P_{\tau \max}(Y)$, где

$$\left. \begin{aligned} P_{\tau \min}(Y) &= \max(P_{\tau \min}, Y - \sum_{i=1}^{\tau-1} P_{i \max}); \\ P_{\tau \max}(Y) &= \min(P_{\tau \max}, Y - \sum_{i=1}^{\tau-1} P_{i \min}). \end{aligned} \right\} \quad (8)$$

Выражения вида $\max(a, b)$, $\min(a, b)$ в правой части (8) следует понимать так:

$$\max(a, b) = \begin{cases} a, & a \geq b \\ b, & a < b \end{cases}; \quad \min(a, b) = \begin{cases} a, & a \leq b \\ b, & a > b \end{cases}.$$

Выражение (7), позволяющее определять значения функции Беллмана $B(Y, \tau)$ по значениям $B(Y, \tau-1)$ этой функции в других точках, называется рекуррентным соотношением Беллмана.

Обратный ход МДП состоит в последовательном определении с помощью функции $F(Y, \tau) = P_{\tau}^{\text{опт}}(Y)$ искомых оптимальных мощностей агрегатов при заданной суммарной нагрузке станции P , причем порядок их нахождения будет обратным нумерации агрегатов, принятой при выполнении прямого хода:

$$P_n = F(P, n) = P_n^{\text{опт}}(P), P_{n-1} = F(P - P_n^{\text{опт}}, n-1) = P_{n-1}^{\text{опт}}(P - P_n^{\text{опт}}), \dots,$$

$$P_1 = F(P - \sum_{i=2}^n P_i, 1) = P_1^{\text{опт}}(P - \sum_{i=2}^n P_i).$$

Рассмотрим подробно порядок действий при построении функций $B(Y, \tau)$, $F(Y, \tau) = P_T^{\text{опт}}(Y)$ и обратный ход МДП при оптимальном распределении заданной суммарной активной мощности P станции между двумя ее агрегатами. Расходные характеристики агрегатов показаны на рис. 7.

Для выполнения прямого хода МДП необходимо произвести следующие действия.

1. Строятся зависимости: $B(Y, 1)$ (рис. 8, а), совпадающая с расходной характеристикой первого агрегата, и $P_1^{\text{опт}}(Y)$, представляющая собой отрезок биссектрисы первого квадранта на интервале изменения абсциссы (рис. 8, б).

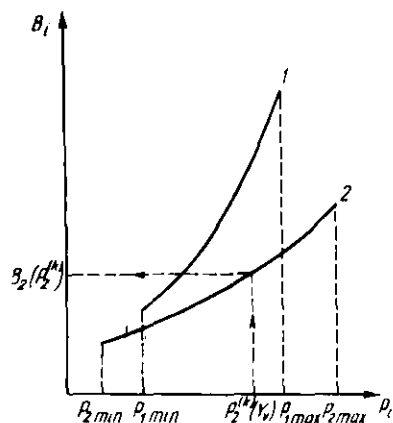


Рис. 7

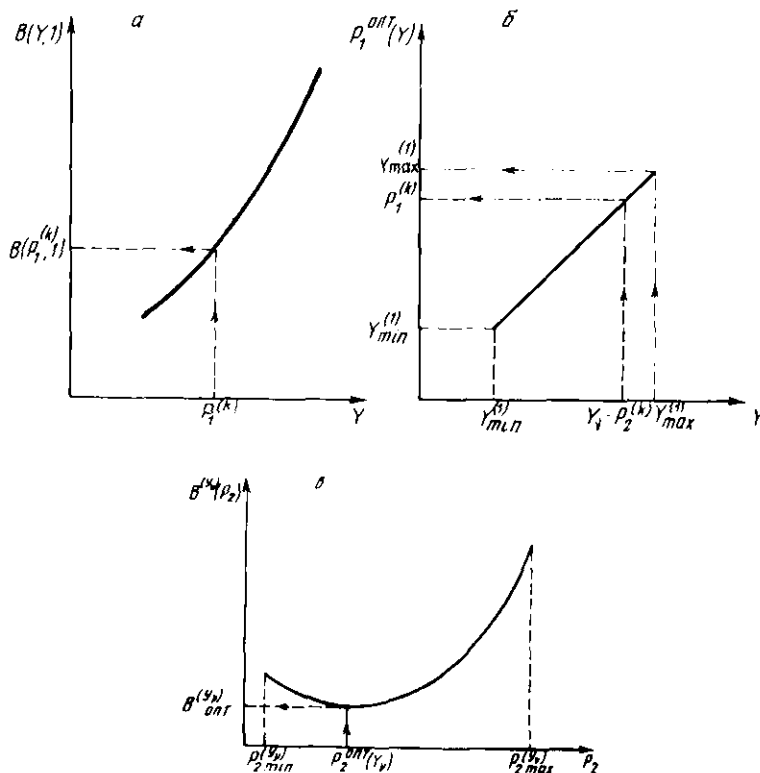


Рис. 8.

$$Y_{\min}^{(1)} \leq Y \leq Y_{\max}^{(1)}; Y_{\min}^{(1)} = P_{1 \min}^{(1)}, Y_{\max}^{(1)} = P_{1 \max}^{(1)}.$$

2. Определяется интервал изменения независимой переменной Y для построения зависимостей $B(Y, 2)$ и $P_2^{\text{опт}}(Y)$:

$$Y_{\min}^{(2)} \leq Y \leq Y_{\max}^{(2)}, Y_{\min}^{(2)} = P_{1 \min}^{(2)} + P_{2 \min}^{(2)}, Y_{\max}^{(2)} = P_{1 \max}^{(2)} + P_{2 \max}^{(2)} \quad (9)$$

и задается набор равноотстоящих значений Y , для которых должны быть определены ординаты указанных зависимостей:

$$Y_1 = Y_{\min}, Y_2 = Y_1 + \Delta Y, \dots, Y_{\nu+1} = Y_{\nu} + \Delta Y, \dots, Y_l = Y_{\max},$$

где ΔY — шаг изменения Y , достаточно малый для того, чтобы обеспечить требуемую точность представления функций $B(Y, 2)$, $P_2^{\text{опт}}(Y)$ на интервале (9).

3. Для каждого значения Y_{ν} определяются ординаты $B(Y_{\nu}, 2)$, $P_2^{\text{опт}}(Y_{\nu})$ искомым зависимостей. Порядок действий при их нахождении следующий:

3.1) определяется интервал изменения независимой переменной P_2 :

$$\begin{aligned} P_{2 \min}^{(Y_{\nu})} &\leq P_2 \leq P_{2 \max}^{(Y_{\nu})}; \\ P_{2 \min}^{(Y_{\nu})} &= \max(P_{2 \min}, Y_{\nu} - P_{1 \max}); \\ P_{2 \max}^{(Y_{\nu})} &= \min(P_{2 \max}, Y_{\nu} - P_{1 \min}) \end{aligned}$$

и набор равноотстоящих значений P_2 на этом интервале

$$\begin{aligned} P_2^{(1)} &= P_{2 \min}^{(Y_{\nu})}, P_2^{(2)} = P_2^{(1)} + \Delta P_2, \dots, P_2^{(k+1)} = P_2^{(k)} + \Delta P_2, \dots, \\ P_2^{(\nu)} &= P_{2 \max}^{(Y_{\nu})}, \end{aligned}$$

где ΔP_2 — выбранный достаточно малый шаг изменения P_2 ;

3.2) для каждого значения $P_2^{(k)}$ последовательно определяются величины $B_2(P_2^{(k)})$ (см. рис. 7), $P_1^{(k)} = Y_{\nu} - P_2^{(k)}$, $B(P_1^{(k)}, 1)$ (рис. 8, б, а), с помощью которых вычисляется ордината

$$B^{(Y_{\nu})}(P_2^{(k)}) = B_2(P_2^{(k)}) + B(P_1^{(k)}, 1)$$

точки зависимости $B^{(Y_{\nu})}(P_2)$, соответствующая значению абсциссы $P_2 = P_2^{(k)}$.

В результате строится график $B^{(Y_{\nu})}(P_2)$ рис. 8, в);

3.3) по построенной в п. 3.2 зависимости $B^{(Y_{\nu})}(P_2)$ отыскивается значение $P_2^{\text{опт}}$, минимизирующее эту функцию (рис. 8, в), которое и представляет собой искомую ординату зависимости $P_2^{\text{опт}}(Y)$, соответствующую абсциссе $Y = Y_{\nu}$ (см. начало п. 3);

3.4) определяется величина $B_{2 \text{опт}}^{(Y_{\nu})}$ как ордината точки зависимости $B_2(P_2)$, соответствующая абсциссе $P_2 = P_2^{\text{опт}}(Y_{\nu})$ (см. рис. 8, в);

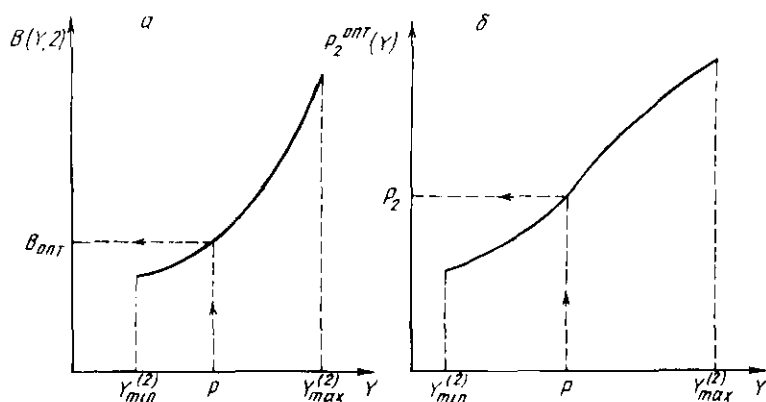


Рис. 9.

3.5) вычисляется значение искомой ординаты зависимости $B(Y, 2)$, соответствующее абсциссе $Y = Y_v$ (см. начало п. 3):

$$B(Y_v, 2) = B_{2\text{опт}}^{(Y_v)} + B(P_1^{(k)}, 1),$$

где $P_1^{(k)} = Y_v - P_2^{\text{опт}}(Y_v)$.

Построением всех точек зависимостей $B(Y, 2)$, $P_2^{\text{опт}}(Y)$ для набора значений независимой переменной (9) заканчивается прямой ход МДП для данного примера (рис. 9, а, б соответственно).

Порядок действий на обратном ходе следующий.

1. С помощью зависимости $P_2^{\text{опт}}(Y)$ определяется искомая оптимальная мощность второго агрегата $P_2 = P_2^{\text{опт}}(P)$ (см. рис. 9, б).

2. По зависимости $P_1^{\text{опт}}(Y)$ вычисляется искомая оптимальная мощность первого агрегата $P_1 = P_1^{\text{опт}}(P - P_2)$.

3. С помощью зависимости $B(Y, 2)$ устанавливается расход топлива на станции при оптимальном распределении суммарной мощности P между агрегатами (см. рис. 9, а)

$$B_{\text{опт}} = B(P, 2).$$

Эту величину, разумеется, можно определить и по расходным характеристикам агрегатов

$$B_{\text{опт}} = f_1(P_1) + f_2(P_2),$$

где P_1, P_2 — найденные на обратном ходе МДП оптимальные нагрузки агрегатов.

Как видно из приведенного примера, вычисления при использовании МДП слишком громоздки для того, чтобы решать практические задачи оптимизации без применения ЭВМ. Поэтому решение задачи (1) — (3) в данной работе выполняется с помощью специальной программы для ЕС ЭВМ.

2. РЕШЕНИЕ ЗАДАЧИ ОПТИМАЛЬНОГО РАСПРЕДЕЛЕНИЯ ЗАДАННОЙ АКТИВНОЙ МОЩНОСТИ ЭЛЕКТРОСТАНЦИИ МЕЖДУ АГРЕГАТАМИ НА ЕС ЭВМ

Описание задачи

Для фиксированного состава включенных в работу агрегатов электростанции с известными расходными характеристиками каждого агрегата решается задача (1)–(3) оптимального распределения заданной суммарной активной мощности станции и выдается на печать значение оптимальной мощности каждого агрегата.

Программа решения задачи

Описание. Программа написана на языке программирования ФОРТРАН IV и состоит из основной программы RASNAGA1 и подпрограммы DYNPRG, имеющей четыре точки входа с именами DYNPRG, ZATROP, MOSHOP, VIVOD. Блок-схема программы показана на рис. 10. Тексты исходных модулей программы RASNAGA1 и подпрограммы DYNPRG приводятся в приложении.

Возможности. Допускается n включенных в работу агрегатов на станции с заданием их расходных характеристик наборами координат $(P_i^{(v)}, B_i^{(v)})$ точек (v – номер точки; i – эксплуатационный набор агрегата; предельное чис-

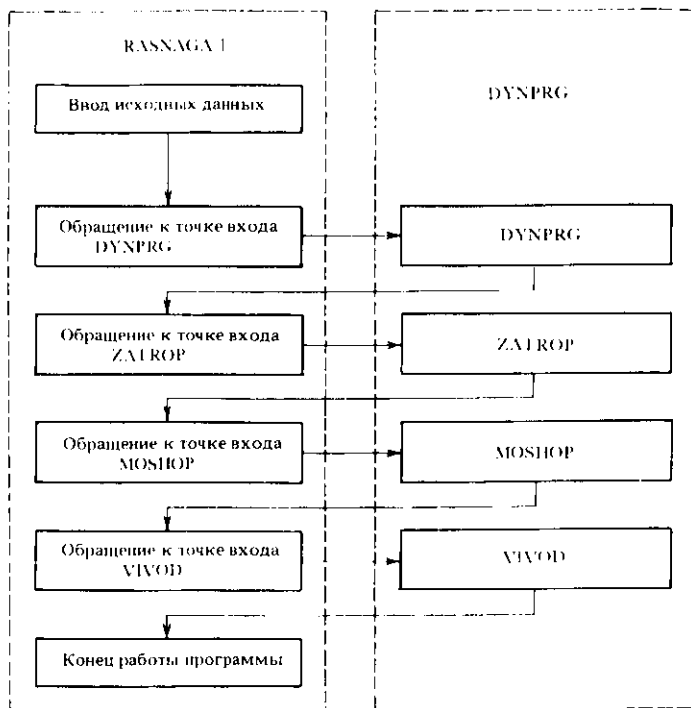


Рис. 10.

ло агрегатов $n = 20$). Число точек каждой расходной характеристики не должно превышать 50. Программа воспринимает расходные характеристики в виде ломаных линий, концами прямолинейных участков которых служат заданные наборы точек. Максимальное количество значений независимой переменной Y при построении функции Беллмана равно 200. Обеспечиваются следующие возможности предельно достижимой точности решения задачи (которая определяется величинами шагов ΔY и ΔP_{τ} на прямом ходе МДП):

1. Должны быть заданы два параметра ΔY^* (в %) и $\Delta Y_{\text{max доп}}$ (в МВт), по которым программа определяет шаг

$$\Delta Y = \min \left(\Delta Y^* \frac{\sum_{i=1}^n P_{i \max}}{100}, \Delta Y_{\text{max доп}} \right).$$

2. Допускаются любые величины ΔY^* , $\Delta Y_{\text{max доп}}$, удовлетворяющие условиям

$$\Delta Y^* \geq \frac{100}{199} \frac{\sum_{i=1}^n (P_{i \max} - P_{i \min})}{\sum_{i=1}^n P_{i \max}}; \quad \Delta Y_{\text{max доп}} \geq \frac{\sum_{i=1}^n (P_{i \max} - P_{i \min})}{199}.$$

3. Должны быть заданы два параметра ΔP_{τ}^* (в %) и $\Delta P_{\tau \text{max доп}}$ (в МВт) произвольной величины, по которым программа определяет шаг:

$$\Delta P_{\tau} = \min \left(\Delta P_{\tau}^* \frac{\max_{i=1, 2, \dots, n} P_{i \max}}{100}, \Delta P_{\tau \text{max доп}} \right).$$

Допускается задание ординат B_i расходных характеристик в любых, но одинаковых для всех агрегатов единицах измерения.

Контрольный пример

Для иллюстрации использования программы RASNAGA1 при решении задачи (1)–(3) внутростанционной оптимизации, а также форм входной и выходной информации, предусмотренных этой программой, в работе приведено решение контрольного (тестового) примера. Исходная информация по контрольному примеру дана в таблицах, иллюстрирующих карты управления и данных программы RASNAGA1.

Ввод

Карты управления. В колоду перфокарт исходной информации при решении задачи должны помещаться следующие карты управления: карта ККА, т.е. количества включенных в работу агрегатов станции (табл. 1); карта КЕИ, т.е. единицы измерения расхода топлива (табл. 2); карта КТТ, т.е. требования к точности решения задачи (табл. 3).

Таблица 1

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Количество агрегатов	12
4-79	Пробелы	
80	Порядковый номер перфокарты в колоде	1

Таблица 2

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-16	Единица измерения	ТОНН_УСЛ_ТОПЛИВА
17-19	Пробелы	
80	Порядковый номер перфокарты в колоде	2

Таблица 3

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-9	ΔY^*	1.00
10-18	$\Delta Y_{\text{max доп}}$	5.00
19-27	ΔP^*	2.00
28-36	$\Delta P_{\tau \text{ max доп}}$	2.00
37-78	Пробелы	
79-80	Порядковый номер перфокарты в колоде	29

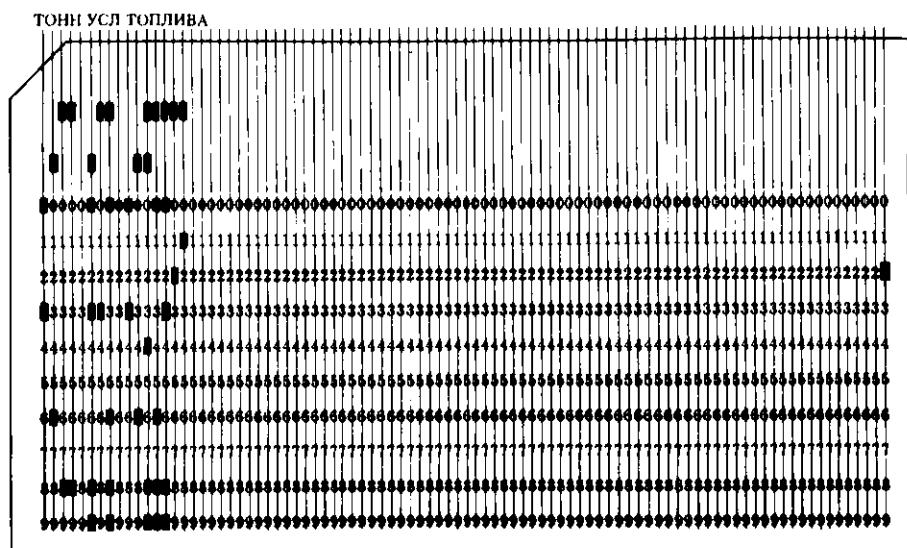


Рис. 11.

Примечания. 1. Полностью исходные данные для контрольного примера будут приведены в таблицах карт управления и данных.

2. Знак — — это условное обозначение пробела. В действительности в соответствующей колонке перфокарты никаких символов не перфорируется.

3. Точка отделяет дробную часть числа от целой.

4. Вид перфокарты, содержащей управляющую информацию КЕИ (см. табл. 3), приведен на рис. 11.

Карты данных. Набор карт данных включает:

1) одну или две карты предварительной информации о включенных в работу агрегатах станции КПИА1 (табл. 4) и КПИА2 (табл. 5), которые содержат эксплуатационные номера агрегатов и количества прямолинейных участков расходной характеристики каждого агрегата, причем карта КПИА2 должна помещаться в колоду лишь при количестве включенных в работу агрегатов $n > 13$.

Примечание. В контрольном примере карта КПИА2 отсутствует.

2) набор карт КРХА расходных характеристик агрегатов, состоящий из поднаборов КРХА_i для каждого агрегата, причем расположение поднаборов в колоде соответствует расположению номеров агрегатов на картах КПИА. Число карт в поднаборе КРХА_i определяется как

$$K_i = \begin{cases} \frac{L_i + 1}{3}, & \text{если } \frac{L_i + 1}{3} - \text{целое число;} \\ E\left(\frac{L_i + 1}{3}\right) + 1, & \text{если } \frac{L_i + 1}{3} - \text{дробное,} \end{cases}$$

Таблица 4

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Номер агрегата	  3
4-6	Количество участков характеристики	  4
7-9	Номер агрегата	  4
10-12	Количество участков характеристики	  4
...	...	...
73-75	Номер агрегата	  
76-78	Количество участков характеристики	  
79-80	Порядковый номер перфокарты в колоде	 3

Таблица 5

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Номер агрегата	
4-6	Количество участков характеристики	
7-9	Номер агрегата	
10-12	Количество участков характеристики	
...	...	
37-39	Номер агрегата	
40-42	Количество участков характеристики	
43-78	Пробелы	
79-80	Порядковый номер перфокарты в колоде	

где L_i — количество участков расходной характеристики i -го агрегата, а запись $E(x)$ означает целую часть (антье) числа x , т.е. наибольшее целое число, такое, что $E(x) \leq x$.

В табл. 6, 7 показана структура первой и последней карт подбора КРХА для первого агрегата ($i = 1$) из контрольного примера;

3) карту заданной активной мощности станции КЗМС (табл. 8).

Таблица 6

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-7	Технический минимум нагрузки	10.
8-22	Расход топлива	2.00
23-29	Значение мощности в порядке возрастания	12.
30-44	Расход топлива	2.40
45-51	Значение мощности в порядке возрастания	14.
52-66	Расход топлива	2.80
67-78	Пробелы	
79-80	Порядковый номер перфокарты в колоде	4

Таблица 7

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-7	Значение мощности в порядке возрастания	17.
8-22	Расход топлива	4.30
23-29	Значение мощности в порядке возрастания	20.
30-44	Расход топлива	5.80
45-51	Значение мощности в порядке возрастания	
52-66	Расход топлива	
67-78	Пробелы	
79-80	Порядковый номер перфокарты в колоде	5

Таблица 8

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-12	Заданная мощность станции	5 60.
13-78	Пробелы	
79-80	Порядковый номер перфокарты в колоде	30

Колода перфокарт. На рис. 12 показано, в каком порядке должны помещаться карты управления и карты данных в колоду перфокарт для решения задачи на ЕС ЭВМ. Содержимое вводимых карт для контрольного примера приведено на рис. 13.

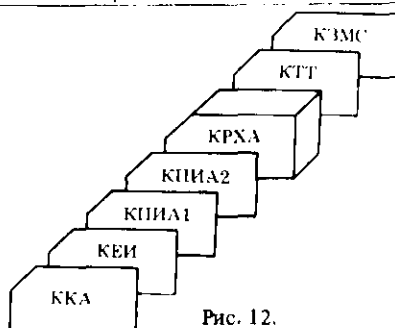


Рис. 12.

// ЛОВ АГРЕГАТЫ ОПТИМИЗАЦИЯ НАГРУЗОК МЕЖАУ АГРЕГАТАМИ СТАНЦИИ
// ЕХЕС НАСНАГА

10.03.17

РАСХОД ТОПЛИВА: 104.0030000 УСЛ ТОПЛИВА

НОМЕР АГРЕГАТА	3	4	7	1	9	2	11	12	3	6
НАГРУЗКА (МВт)	10	60	31	39	50	70	34	35	70	60

НОМЕР АГРЕГАТА	10	8
НАГРУЗКА (МВт)	70	30

ЕОJ АГРЕГАТЫ РЕШЕНИЕ ЗАДАЧ ПО ВАРИАНТУ МВ ВЫПОЛНЕНО

10.04.09 . DURATION 00.00.52

Рис. 14.

Вывод

На печать выводятся следующие результаты решения задачи:

1) расход топлива на станции при оптимальном распределении заданной суммарной мощности между агрегатами;

2) функция $P(i)$, где i — эксплуатационный номер агрегата; P — оптимальная активная мощность агрегата при заданной мощности станции.

Данные печатаются в удобном для восприятия виде. Результаты решения контрольного примера показаны на рис. 14.

Инструкция по применению программы

С целью сокращения затрат машинного времени программа каталогизирована в личную библиотеку абсолютных модулей и постоянно хранится на магнитном диске. Для настройки операционной системы на решение задачи с помощью описываемой программы колода перфокарт, показанная на рис. 12 (вводной файл данных), должна быть дополнена картами управляющих операторов операционной системы. Структура карт управляющих операторов показана в табл. 9–12.

Таблица 9

Колонка перфокарты	Содержимое	Назначение содержимого
1–7 8–16 17–63 64–80	//JOB АГРЕГАТЫ 'ОПТИМИЗАЦИЯ НАГРУЗОК МЕЖДУ АГРЕГАТАМИ СТАНЦИИ' Пробелы	Код оператора "Начало задания" Имя задания Комментарий

Таблица 10

Колонка перфокарты	Содержимое	Назначение содержимого
1–8 9–16 17–80	//EXEC RASNAGA1 Пробелы	Код оператора "Выполнить программу" Имя программы

Таблица 11

Колонка перфокарты	Содержимое	Назначение содержимого
1–2 3–80	/* Пробелы	Код оператора "Конец файла данных"

Таблица 12

Колонка перфокарты	Содержимое	Назначение содержимого
1-13	/&	Код оператора "Конец задания"
14-51	РЕШЕНИЕ ЗАДАЧИ ПО ВАРИАНТУ N ВЫПОЛНЕНО	Комментарий
52-80	Пробелы	

Размещение карт управляющих операторов в колоде перфокарт задания показано на рис. 15 (аббревиатура ККУД на рис. 15 означает: "Колода карт управления и данных"). Состав ККУД приведен на рис. 12

Для решения задачи на ЭВМ в режиме операторного счета колода перфокарт, сформированная в соответствии с рис. 15, должна быть сдана оператору ЭВМ для установки в поток заданий к заранее запланированному моменту времени.

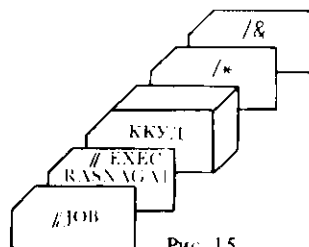


Рис. 15.

Примечание. В позицию 42 карты /& (см. табл. 12, рис. 15) заносится номер заданного преподавателем варианта исходных данных (см. раздел 6 данной лабораторной работы).

Сообщения об ошибках и время решения

За время работы программы могут быть выданы сообщения об ошибках в случаях, предусмотренных операционной системой ДОС/ЕС-2.1. Смысл сообщения можно установить из тома документации к ДОС/ЕС "Сообщения системы. Руководство для оператора и программиста".

Быстродействие программы можно охарактеризовать временем решения контрольного примера, которое составляет 52 с.

Подпрограмма DYNPRG

Назначение. Оптимальное распределение нагрузок между совместно работающими объектами (агрегаты, станции), для которых заданы расходные характеристики. Назначение участков, соответствующих различным точкам входа:

- DYNPRG — выполнение прямого хода МДП;
- ZATROP — расчет оптимальной величины расхода топлива для заданной суммарной нагрузки;
- MOSHOP — определение оптимальных нагрузок объектов для заданной суммарной нагрузки;
- VIVOD — печать результатов решения задачи.

Обращение. а) CALL DYNPRG (N, A, KOLUCH, IZLOM, SHAGY, MSHAGY, SHAGP, MSHAGP, OTVET, KOLZNY, HAR);

- б) CALL ZATROP (PSUMZD, IZLOM);
- в) CALL MOSHOP (PSUMZD, IZLOM, HAR, OTVET);
- г) CALL VIVOD (EDIZM, TEXT, OTVET, NOMAGR).

Описание параметров. В обращениях: N — количество объектов; A — имя массива, содержащего числа участков расходных характеристик объектов; KOLUCH — величина $2(l_{\max} + 1)$, где $l_{\max}$ — наибольшее количество участков расходной характеристики среди всех объектов; IZLOM — имя двумерного массива, содержащего координаты точек расходных характеристик и имеющего предельные значения измерений соответственно KOLUCH и N; SHAGY — величина ΔY^* ; MSHAGY — величина $\Delta Y_{\max \text{ доп}}$; SHAGP — величина ΔP^* ; MSHAGP — величина $\Delta P_{\max \text{ доп}}$; OTVET — имя массива оптимальных нагрузок объектов; KOLZNY — максимальное количество значений независимой переменной Y при построении функции Беллмана; HAR — двумерный рабочий массив с предельными значениями измерений соответственно KOLZNY и N; PSUMZD — заданная суммарная нагрузка; EDIZM — имя массива, содержащего текстовую константу длиной 16 символов, указывающую единицу измерения расхода топлива; TEXT — имя массива, содержащего текстовую константу длиной 8 символов, указывающую наименование объекта в родительном падеже ('АГРЕГАТА' или 'СТАНЦИИ'); NOMAGR — имя массива эксплуатационных номеров объектов.

Примечание. Четыре точки входа организованы для того, чтобы можно было использовать подпрограмму в новых программах, которые в отличие от программы RASNAGAI требуют обращения к различным участкам подпрограммы DYNPRG неодинаковое число раз.

Метод. Метод решения задачи описан в разделе "Краткие теоретические сведения". На рис. 16 приводится схема подпрограммы DYNPRG, представленная в виде четырех частей (a—г). Каждая часть соответствует отдельной точке входа подпрограммы.

Примечание. Обозначение $X \Leftarrow Y$ на схемах рис.16 означает, что величине (набору величин) X присваивается значение величины (набора величин) Y.

3. ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

В данной лабораторной работе ставится задача выяснения условий применимости методов удельных приростов и динамического программирования к решению задач (1)–(3). С этой целью студентам предлагается решить указанную задачу при конкретных исходных данных вручную методом удельных приростов и по программе RASNAGAI для ЕС ЭВМ — методом динамического программирования. Таким образом, при подготовке к работе необходимо:

1) изучить раздел 1 данной лабораторной работы, обратив особое внимание на описание алгоритма решения задачи (1)–(3) методом удельных приростов;

2) изучить раздел 2, обратив особое внимание на правила подготовки исходных данных для решения задачи (1)–(3) с помощью программы RASNAGAI;

3) получить у преподавателя один из вариантов исходных данных, приведенных в параграфе 6;

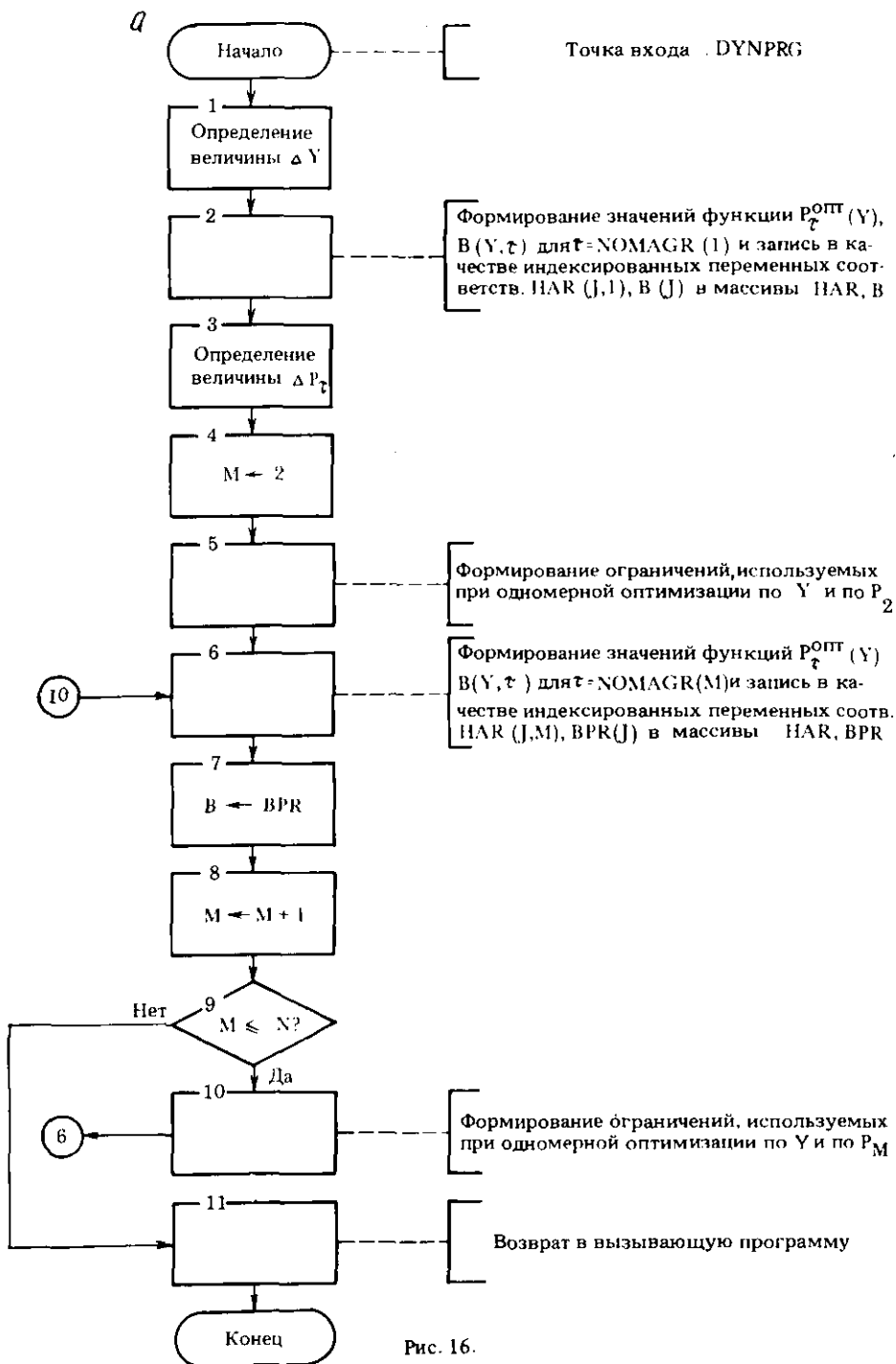


Рис. 16.

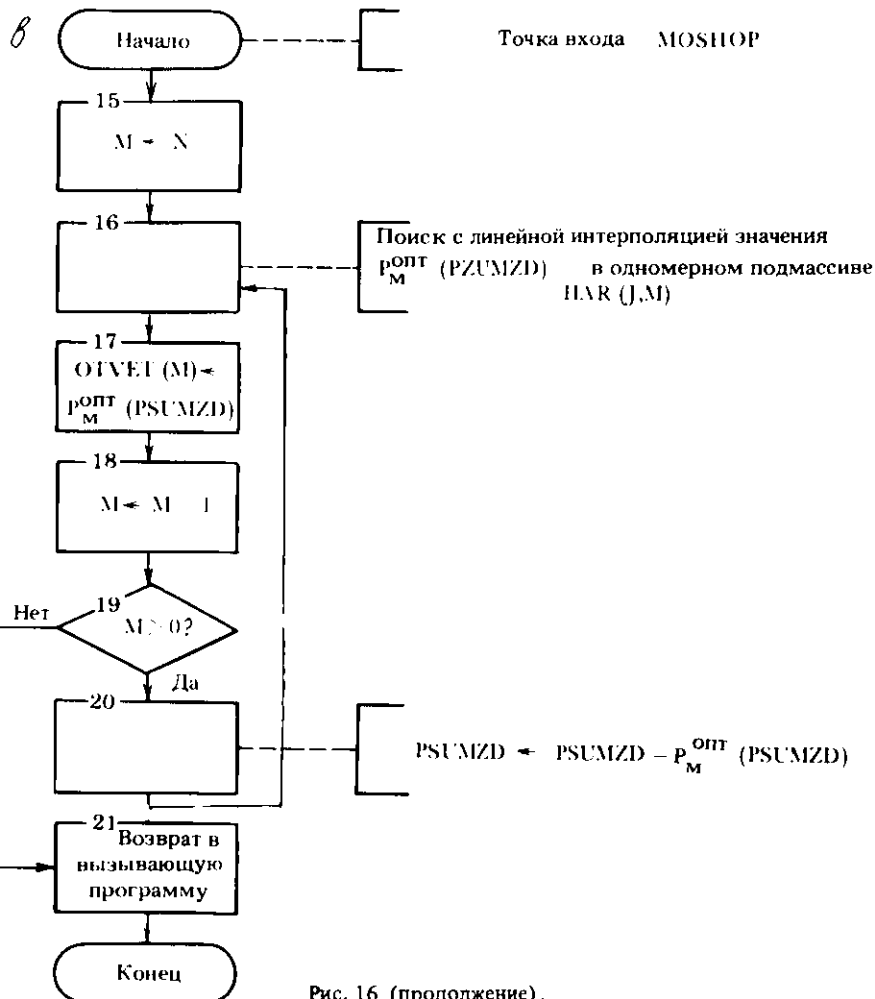
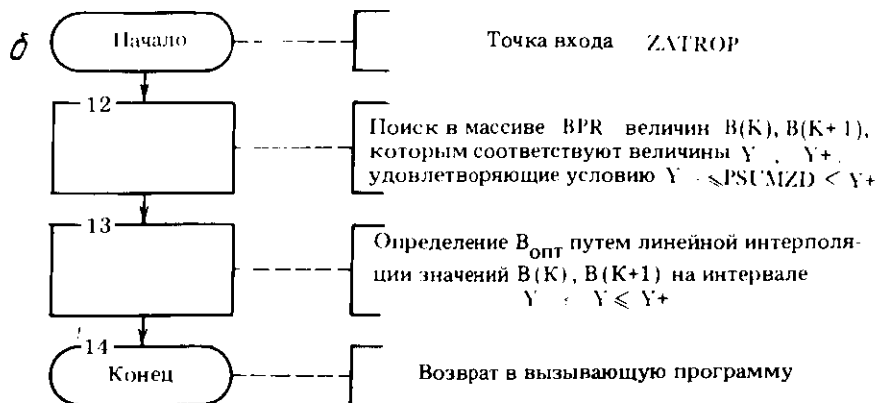
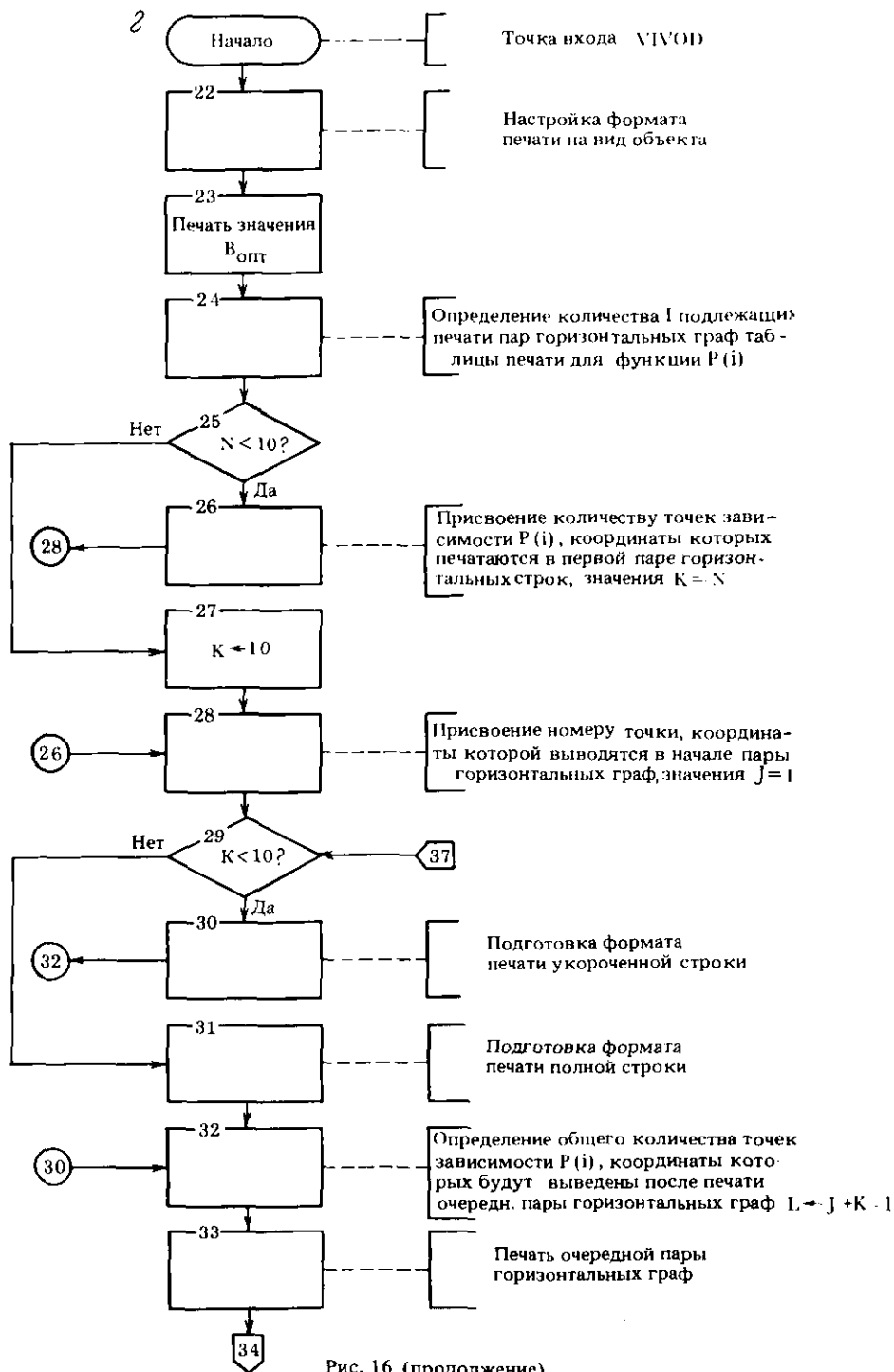


Рис. 16 (продолжение).



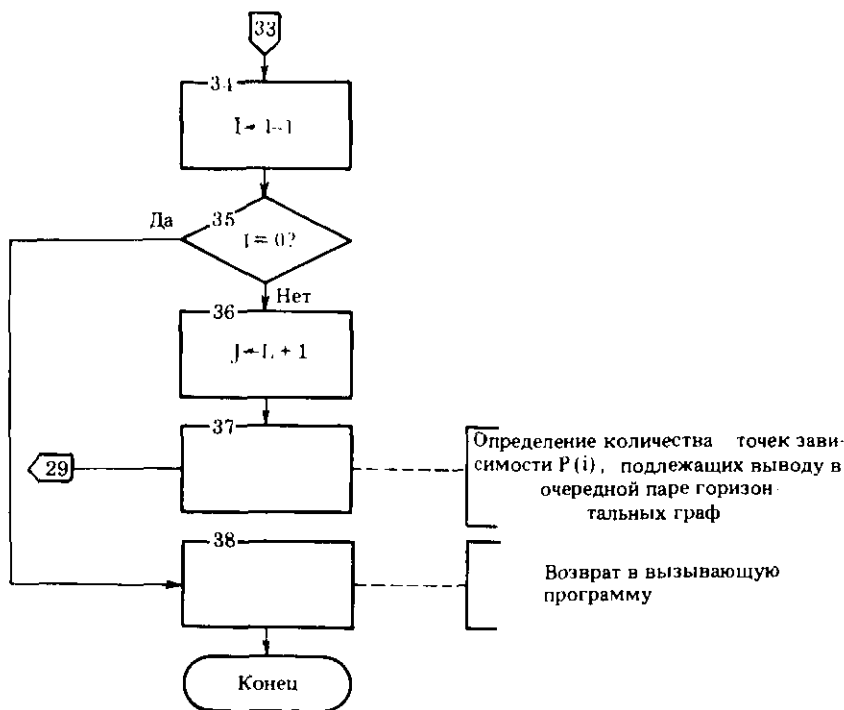


Рис. 16 (окончание).

- 4) решить задачу (1) – (3) вручную методом удельных приростов;
- 5) подготовить исходную информацию на бланках по предложенному варианту задания для сдачи на перфорацию.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать следующие разделы:

- 1) цель работы;
- 2) характеристику условий применимости методов удельных приростов и динамического программирования к решению задачи оптимизации активных мощностей на электростанции;
- 3) распечатку исходных данных;
- 4) оформленное в виде таблиц и графиков решение задачи методом удельных приростов;
- 5) распечатку результатов решения задачи методом динамического программирования;
- 6) сопоставление результатов решения задачи двумя методами с объяснением причин расхождений, если таковые имеются.

П р и м е ч а н и е. Распечатка исходных данных по каждому варианту выполняется с помощью специального задания для ДОС/ЕС-2.1, подготовленного преподавателем. Правила составления этого задания даны в лабораторной работе 2.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В каких случаях правило равенства удельных приростов является необходимым и достаточным условием получения решения задачи (1)–(3), а в каких – только необходимым? В чем состоит различие понятий "необходимое" и "достаточное"?
2. Объясните физический смысл независимых переменных функции Беллмана для задачи (1)–(3).
3. Для чего служит рекуррентное соотношение Беллмана при решении задачи методом динамического программирования?
4. Каков смысл принципа оптимальности Беллмана?
5. Объясните разницу между понятиями локального и глобального минимумов функции применительно к целевой функции задачи (1)–(3).
6. Укажите признак выпуклости функции.
7. Как наглядно проиллюстрировать неприменимость метода удельных приростов к решению задачи (1)–(3) в случае невыпуклых расходных характеристик?
8. Всегда ли метод равенства удельных приростов является универсальным?
9. Что такое критерий оптимальности?
10. Как зависит выбор метода оптимизации от вида характеристик удельных приростов электрических станций?
11. В каких случаях принцип равенства удельных приростов непригоден для решения задачи оптимизации?
12. Чем определяется целевая функция и ограничения задачи оптимизации?
13. Почему отступление от правила удельных приростов приводит к перерасходу топлива?
14. Как сформировать колоду перфокарт для решения задачи оптимизации?

6. ВАРИАНТЫ ИСХОДНЫХ ДАННЫХ

Вариант 1

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$
9	75	27,9	86	29,2	97	31,5	110	33,7	120	36,8	132	40,9	143	49,6	58,6
2	68	27,8	78	30,6	101	36,0	127	47,3	150	58,4					
4	50	21,4	57	22,5	65	24,2	73	26,0	80	28,3	88	31,5	95	38,2	45,0
8	46	19,7	71	29,5	81	37,4	100	45,0							
6	45	21,4	52	23,6	67	27,9	85	36,4	100	45,0					

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тонн камен.угля; $\Delta Y^* = 0,5\%$; $\Delta Y_{\text{max доп}} = 2 \text{ MBT}$; $P = 450 \text{ MBT}$.

Вариант 2

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$
3	75	51,0	86	53,6	97	57,6	110	61,7	120	67,4	132	75	143	91	107,1
1	68	51,0	78	56,1	101	66,4	127	86,8	150	107,0					
5	69	46,5	106	70,0	122	88,4	150	107,0							
7	50	39,2	57	41,2	65	43,3	73	47,5	80	51,8	88	57,6	95	70	82,4
4	52	43,2	67	51,0	85	66,7	100	82,4							

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тонн бурого угля; $\Delta Y^* = 0,7\%$; $\Delta Y_{\text{max доп}} = 2,5 \text{ MBT}$; $P = 550 \text{ MBT}$.

Вариант 3

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
1	100	27,1	114	28,5	130	30,7	146	32,8	160	35,8	176	39,8	190	48,4	200	57,0
6	90	27,2	104	30,0	134	35,4	170	46,3	200	50,7						
2	50	16,9	57	17,8	65	19,2	73	20,5	80	22,4	88	24,9	95	30,2	100	35,6
5	45	16,9	52	18,6	67	22,0	85	28,8	100	35,6						
8	46	15,5	71	23,2	81	29,4	100	35,6								

У к а з а н и е. Задание выполнить при условиях: единица измерения B – куб. м газа; $\Delta Y^* = 0,6\%$; $\Delta Y_{\max \text{ доп}} = 3 \text{ МВт}$; $P = 400 \text{ МВт}$.

Вариант 4

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
10	150	29,7	171	31,2	195	33,6	219	36,0	240	39,2	264	43,6	285	52,9	300	62,4
1	135	29,8	156	32,7	201	38,7	255	50,6	300	62,4						
5	138	27,0	212	40,6	244	51,5	300	62,4								
8	50	12,5	57	13,1	65	14,0	73	15,1	80	16,5	88	18,4	95	22,2	100	26,2
4	45	12,5	52	13,7	67	16,2	85	22,2	100	26,2						
3	46	11,4	71	17,2	81	21,6	100	26,2								

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тонн мазута; $\Delta Y^* = 1,0\%$; $\Delta Y_{\max \text{ доп}} = 5 \text{ МВт}$; $P = 1000 \text{ МВт}$.

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
1	150	1275	171	1340	195	1440	219	1545	240	1685	264	1875	285	2270	300	2680
5	135	1270	156	1400	201	1660	255	2170	300	2680					100	1140
8	50	542	57	570	65	614	73	656	80	715	88	797	95	965		
11	45	540	52	596	67	705	85	920	100	1140						
3	46	495	71	743	81	942	100	1140								
2	45	540	52	595	67	705	85	920	100	1140						

У к а з а н и е. Задание выполнить при условиях: единица измерения B – руб; $\Delta Y^* = 0,4\%$; $\Delta Y_{\text{max доп}} = 4,5 \text{ МВт}$; $P = 750 \text{ МВт}$.

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
7	150	48,7	171	51,1	195	55,1	219	59,0	240	64,4	264	71,6	285	86,5	300	102,3
2	135	48,5	156	53,4	201	63,0	255	82,5	300	102,3						
4	138	44,5	212	66,5	244	84,5	300	102,3								
6	100	33,4	114	35,0	130	37,6	146	40,4	160	44,0	176	49,0	190	59,4	200	70,0
3	90	33,3	104	36,7	134	43,4	170	56,6	200	70,0						

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тонн усл. топлива; $\Delta Y^* = 0,6\%$; $\Delta Y_{\text{max доп}} = 5 \text{ МВт}$; $P = 900 \text{ МВт}$.

Вариант 7

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
5	250	0,082	285	0,086	325	0,092	365	0,099	400	0,108	440	0,121	475	0,146	500	0,172
10	225	0,082	260	0,091	335	0,106	425	0,140	500	0,172						
1	230	0,075	355	0,112	405	0,142	500	0,172								
4	225	0,082	260	0,091	335	0,106	425	0,140	500	0,172						
7	130	0,038	160	0,044	190	0,059	200	0,070								
2	90	0,033	104	0,038	134	0,043	170	0,056	200	0,070						

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тыс. тонн усл. топлива; $\Delta Y^* = 1,1\%$; $\Delta Y_{\max \text{ доп}} = 10 \text{ МВт}$; $P = 1800 \text{ МВт}$.

Вариант 8

i	$P_i^{(1)}$	$B_i^{(1)}$	$P_i^{(2)}$	$B_i^{(2)}$	$P_i^{(3)}$	$B_i^{(3)}$	$P_i^{(4)}$	$B_i^{(4)}$	$P_i^{(5)}$	$B_i^{(5)}$	$P_i^{(6)}$	$B_i^{(6)}$	$P_i^{(7)}$	$B_i^{(7)}$	$P_i^{(8)}$	$B_i^{(8)}$
6	400	3,34	450	3,50	520	3,77	580	4,04	640	4,40	705	4,91	760	5,95	800	7,0
1	100	0,875	130	0,99	160	1,154	190	1,56	200	1,836						
2	90	0,875	134	1,140	170	1,48	200	1,836								
4	92	0,795	142	1,20	162	1,51	200	1,83								
5	100	0,875	130	0,99	160	1,154	190	1,56	200	1,836						

У к а з а н и е. Задание выполнить при условиях: единица измерения B – тыс. руб.; $\Delta Y^* = 2,0\%$; $\Delta Y_{\max \text{ доп}} = 6 \text{ МВт}$; $P = 1200 \text{ МВт}$.

П р и м е ч а н и е. Удельный прирост расхода топлива на интервале $P_i^{(v)} \leq P_i \leq P_i^{(v+1)}$ считается неизменным и принимается равным

$$\varepsilon_i^{(v)} = \frac{B_i^{(v+1)} - B_i^{(v)}}{P_i^{(v+1)} - P_i^{(v)}}.$$

ИЗУЧЕНИЕ ВОЗМОЖНОСТЕЙ, ПРЕДОСТАВЛЯЕМЫХ ПОЛЬЗОВАТЕЛЮ ОПЕРАЦИОННОЙ СИСТЕМОЙ ДОС/ЕС НА ПРИМЕРЕ ЗАДАЧИ ОПТИМИЗАЦИИ РАСПРЕДЕЛЕНИЯ НАГРУЗОК МЕЖДУ АГРЕГАТАМИ ЭЛЕКТРОСТАНЦИИ

Цель работы: ознакомиться с основными функциями операционных систем на примере ДОС/ЕС-2.1, назначением основных системных программ, входящих в ее состав, и их наиболее употребительными управляющими операторами на примере программ оптимального распределения активных мощностей между агрегатами электростанций; получить навыки в составлении заданий на трансляцию проблемных программ, редактирование однофазовых и многофазовых программ, каталогизацию их в библиотеки исходных, объектных и абсолютных модулей, распечатку исходных данных при решении задачи на ЭВМ, исполнение рабочей программы.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Основные функции операционных систем ЕС ЭВМ

Каждая ЭВМ третьего поколения снабжается набором так называемых системных программ, постоянно хранящихся на внешнем запоминающем устройстве — резиденте системы (обычно это пакет магнитных дисков). Эти программы, без которых невозможно решение задач на ЭВМ, образуют основу базового математического обеспечения ЭВМ — ее операционную систему. Основные функции последней состоят в следующем.

1. Обеспечивается реализация режима пакетной обработки заданий. Режим пакетной обработки заключается в том, что запросы пользователей, оформленные в виде составленных по определенным правилам заданий, воспринимаются и удовлетворяются машиной (в случае, если возможности ЭВМ это позволяют) автоматически, сменяя друг друга в порядке их появления во входном потоке заданий (последний представляет собой последовательность соответствующих заданиям колод перфокарт, установленных на системном устройстве ввода). Таким образом, если аппаратная реализация самой машины обеспечивает автоматическое выполнение каждой отдельной программы, то операционная система автоматизирует весь процесс нормального функционирования ЭВМ.

2. Операционная система в значительной мере автоматизирует процесс получения пользователем программ, пригодных для непосредственного выполнения машиной. Это достигается благодаря тому, что операционная система способна воспринимать программы, написанные на разных алгоритмических языках.

3. Операционная система позволяет накапливать, хранить и оперативно корректировать большие объемы данных (файлы), состоящие из записей необходимой пользователю структуры, на внешних носителях (магнитных

лентах, магнитных дисках) и легко копировать их на внешних носителях другой физической природы, например выводить на печать.

4. Обеспечивается возможность создания на внешних носителях обширных системных и личных библиотек программ на разных стадиях готовности последних к непосредственному исполнению машиной, благодаря чему сокращается время решения задачи при многократном использовании программы.

5. Представляется возможность создания больших программ при ограниченных ресурсах основной памяти ЭВМ благодаря автоматическому вызову в заданном пользователем порядке нужных частей программы с внешних носителей, куда они были предварительно записаны.

6. Операционная система автоматизирует процесс создания программ, составленных из блоков, хранящихся в библиотеках, а также подготовленных независимо друг от друга пользователем, причем различные блоки могут быть написаны на разных языках программирования и находиться на разных стадиях готовности к непосредственному исполнению машиной.

Организация процесса прохождения заданий в режиме пакетной обработки

Автоматический процесс обработки входного потока заданий осуществляется под управлением двух программ — программы СУПЕРВИЗОР и программы УПРАВЛЕНИЕ ЗАДАНИЯМИ, которые являются основными составными частями УПРАВЛЯЮЩЕЙ ПРОГРАММЫ операционной системы.

С целью уменьшения объема занимаемой основной памяти СУПЕРВИЗОР создается из нескольких частей, одна из которых — ядро СУПЕРВИЗОРА — постоянно находится в основной памяти ЭВМ, а другие — фазы (транзиты СУПЕРВИЗОРА) — вызываются ядром в основную память для исполнения по мере необходимости. Ядро СУПЕРВИЗОРА образует верхнюю ступень иерархии в системе управления потоком заданий. Для обработки каждого очередного задания ядро СУПЕРВИЗОРА вызывает в основную память программу УПРАВЛЕНИЕ ЗАДАНИЯМИ, а при необходимости (в зависимости от типа задания) и те или иные фазы (транзиты СУПЕРВИЗОРА).

Программа УПРАВЛЕНИЕ ЗАДАНИЯМИ вводит в основную память управляющие операторы, содержащиеся в задании, расшифровывает их и сообщает ядру СУПЕРВИЗОРА, какую программу требуется вызвать из библиотеки в соответствии с заданием. После этого супервизор загружает указанную программу на место программы УПРАВЛЕНИЕ ЗАДАНИЯМИ и передает ей управление. Так как задание пользователя может требовать от ЭВМ последовательной работы нескольких программ, чему соответствуют различные шаги задания, описанный процесс повторяется на каждом шаге задания.

Общая характеристика управляющих операторов системных программ

Текст управляющих операторов, сообщаемый программе УПРАВЛЕНИЕ ЗАДАНИЯМИ для настройки операционной системы ДОС/ЕС-2.1 на выполнение предписываемых заданием функций, как и управляющих операторов других программ операционной системы, состоит из двух основных частей: ко-

да операции, определяющего вид информации, сообщаемой оператором, и некоторого числа параметров, конкретные значения которых придают указанной информации характер однозначных предписаний. Если правила записи управляющего оператора допускают возможность указывать или не указывать в тексте этого оператора значение некоторого параметра, последний называется *необязательным*. Такие параметры возможны по двум причинам: 1) до появления во входном потоке рассматриваемого управляющего оператора операционной системе уже было сообщено тем или иным способом значение данного параметра, которое требуется в этом операторе; 2) если данный параметр отсутствует в тексте рассматриваемого управляющего оператора, текст последнего не теряет свойства однозначного предписания, но в меньшей степени изменяет те функции, на которые была настроена операционная система до появления этого оператора.

При описании формата (допускаемой правилами структуры текста) управляющего оператора принимаются две условности: 1) для указания на то, что тот или иной параметр является *необязательным*, он будет заключаться в квадратные скобки, хотя в конкретные тексты управляющих операторов квадратные скобки записываться не должны; 2) указывается наименование параметра, которое в текстах операторов должно заменяться конкретным значением параметра.

При подготовке пользователем задания тексты управляющих операторов, как и остальная информация задания, т.е. данные для решения конкретной задачи и, возможно, программа пользователя (проблемная программа), записываются на стандартных бланках. Содержимое последних переносится построчно на 80-колонокковые перфокарты с помощью специального кода перфорации, при котором каждому символу записи на бланке соответствуют пробивки в одной колонке перфокарты. Формат управляющих операторов, помимо кода операции и параметров, может предусматривать также нанесение на перфокарту специальных пробивок, указывающих на принадлежность оператора к той или иной группе управляющих операторов. В частности, почти все управляющие операторы программы УПРАВЛЕНИЕ ЗАДАНИЯМИ должны иметь в 1-й и 2-й позициях по символу /. Формат всех управляющих операторов допускает, чтобы между кодом операции и значениями параметров на перфокарте имелось любое число пустых колонок (пробелов), но требует, чтобы хотя бы один пробел обязательно был. Параметры должны отделяться друг от друга запятыми, а пробелы между ними обычно не допускаются.

Любую информацию для программы УПРАВЛЕНИЕ ЗАДАНИЯМИ можно сообщить с помощью либо управляющего оператора либо соответствующей директивы. Различие между ними состоит в том, что управляющий оператор действует только в данном задании операционной системе, а директива — вплоть до ее отмены в последующих (или данном) заданиях. Формат директив отличается отсутствием символа / в 1-й и 2-й позициях.

Приведем основные управляющие операторы программы УПРАВЛЕНИЕ ЗАДАНИЯМИ: JOB — начать задание; /& — конец задания; OPTION — установить режим; EXEC — выполнить программу; /* — конец файла.

Физические и логические устройства ввода-вывода

Чтобы обеспечить пользователю возможность написания текста проблемной программы независимо от физической природы и конкретных номеров устройств ввода-вывода на каждой отдельной ЭВМ, операционная система позволяет использовать в программах имена так называемых логических устройств ввода-вывода, которые приняты стандартными для всех возможных моделей ЕС ЭВМ. Затем при подготовке задания для конкретной ЭВМ на исполнение проблемной программы пользователь должен указывать операционной системе соответствие между логическими и физическими устройствами ввода-вывода, необходимыми для исполнения задания, или присвоить конкретные адреса устройствам ввода-вывода. Для этой цели служит управляющий оператор ASSIGN.

В ЕС ЭВМ используются следующие имена логических устройств ввода-вывода: SYSRES – резидент системы, т.е. устройство ввода-вывода (диск-вод с пакетом магнитных дисков), на котором записана сама операционная система; SYSRDR – системное считывающее устройство, т.е. устройство ввода-вывода (ввод перфокарт, накопитель на магнитной ленте (НМЛ) или на магнитных дисках (НМД)), на котором устанавливается входной поток заданий (управляющие операторы); SYSIPT – системный ввод, т.е. устройство ввода-вывода (ввод перфокарт, НМЛ, НМД), с которого вводится исходная информация по заданиям (данные программы), за исключением управляющих операторов; SYSIN – совмещенное системное устройство и системный ввод (ввод перфокарт, НМЛ, НМД), которое может организовываться вместо SYSRDR и SYSIPT; SYSPCH – системный перфоратор (вывод перфокарт, НМЛ, НМД), т.е. устройство ввода-вывода, на которое выводится из ЭВМ информация в виде перфокарт и/или их образов на носителях другой физической природы; SYSLST – системная печать (алфавитно-цифровое печатающее устройство – АЦПУ, НМЛ, НМД), т.е. устройство ввода-вывода, на которое выводится из ЭВМ вся информация, предназначенная для визуального восприятия пользователем, в виде строчек печати либо их образов; SYSLNK – рабочая область программы РЕДАКТОР операционной системы, т.е. область резидентного файла на магнитных дисках, используемая программой РЕДАКТОР при обработке проблемной программы (редактировании); SYSLOG – системное устройство связи ЭВМ и оператора (пишущая машинка, АЦПУ), т.е. устройство, посредством которого осуществляется весь взаимный обмен информацией между ЭВМ и оператором ЭВМ – лицом, управляющим процессом прохождения потока заданий на ЭВМ; SYSSLB – личная библиотека исходных модулей, т.е. устройство ввода-вывода (НМЛ), содержащее библиотеку программ на исходном языке программирования, созданную данным пользователем; SYSRLB – личная библиотека объектных модулей, т.е. устройство ввода-вывода (НМД), содержащее программы пользователя, переведенные (транслированные) операционной системой на язык машинных команд, но еще не подготовленные полностью к использованию (не отредактированные); SYSCLB – личная библиотека абсолютных модулей, т.е. устройство ввода-вывода (НМЛ), содержащее программы пользователя, полностью готовые к исполнению; SYSØ1 SYSØ11 – устройства ввода-вывода, на которых

установлены носители (НМЛ, НМД, перфокарты, печать), содержащие файлы, предназначенные для обмена информацией с основной памятью ЭВМ в процессе выполнения задания.

Контроль правильности исходных данных

При решении конкретной задачи всегда должен осуществляться контроль за правильностью данных, вводимых в ЭВМ. Назначение этого контроля – выявить и устранить ошибки, допущенные при кодировании и перфорации данных. Если время, необходимое для получения ответа по задаче, мало (например, в задачах оперативного управления, данные для которых вводятся в ЭВМ через каналы телекоммуникационного ввода либо с перфоленты, на которую принимаются данные с телетайпа), то контроль данных должен осуществляться программным путем. С этой целью пользователь при написании проблемной программы должен предусмотреть программные тесты диагностики, позволяющие проверить наличие заранее известных свойств данных для соответствующей задачи.

При отсутствии жестких ограничений на допустимое время решения задачи следует проверить каждое данное, вводимое в ЭВМ. Для этого в специальном задании, предшествующем заданию на исполнение программы решения задачи, необходимо произвести ввод и распечатку данных. Для этой цели может использоваться одна из системных программ операционной системы ДОС/ЕС–2.1 – программа перезаписи КАРТЫ-ПЕЧАТЬ/КАРТЫ.

Общая схема (несколько приближенная) возможных этапов обработки задания операционной системой ДОС/ЕС–2.1 показана на рис. 17.

На каждом этапе обработки задания на SYSLST выдаются диагностические сообщения операционной системы, т.е. сообщение об ошибках в программе пользователя, обнаруженных той системной программой, которая обрабатывает программу пользователя в данный момент.

Этапы обработки проблемных программ операционной системой

Составление задания на трансляцию исходных модулей

При составлении задания только на трансляцию пользователь должен сообщить программе УПРАВЛЕНИЕ ЗАДАНИЯМИ посредством ее управляющих операторов следующую информацию:

1) язык программирования, на котором написан каждый исходный модуль, подлежащий трансляции в данном задании;

2) состав визуальной информации о результатах трансляции, которую желает получить пользователь на SYSLST;

3) требуется ли пользователю получить текст объектного модуля на SYSPCH либо целью трансляции является лишь получение информации о синтаксических ошибках в исходном модуле (вместе с информацией об относительных адресах операторов, переменных и массивов текстов исходных модулей или без этой информации);

4) назначение физических устройств ввода-вывода используемым в задании логическим устройствам.

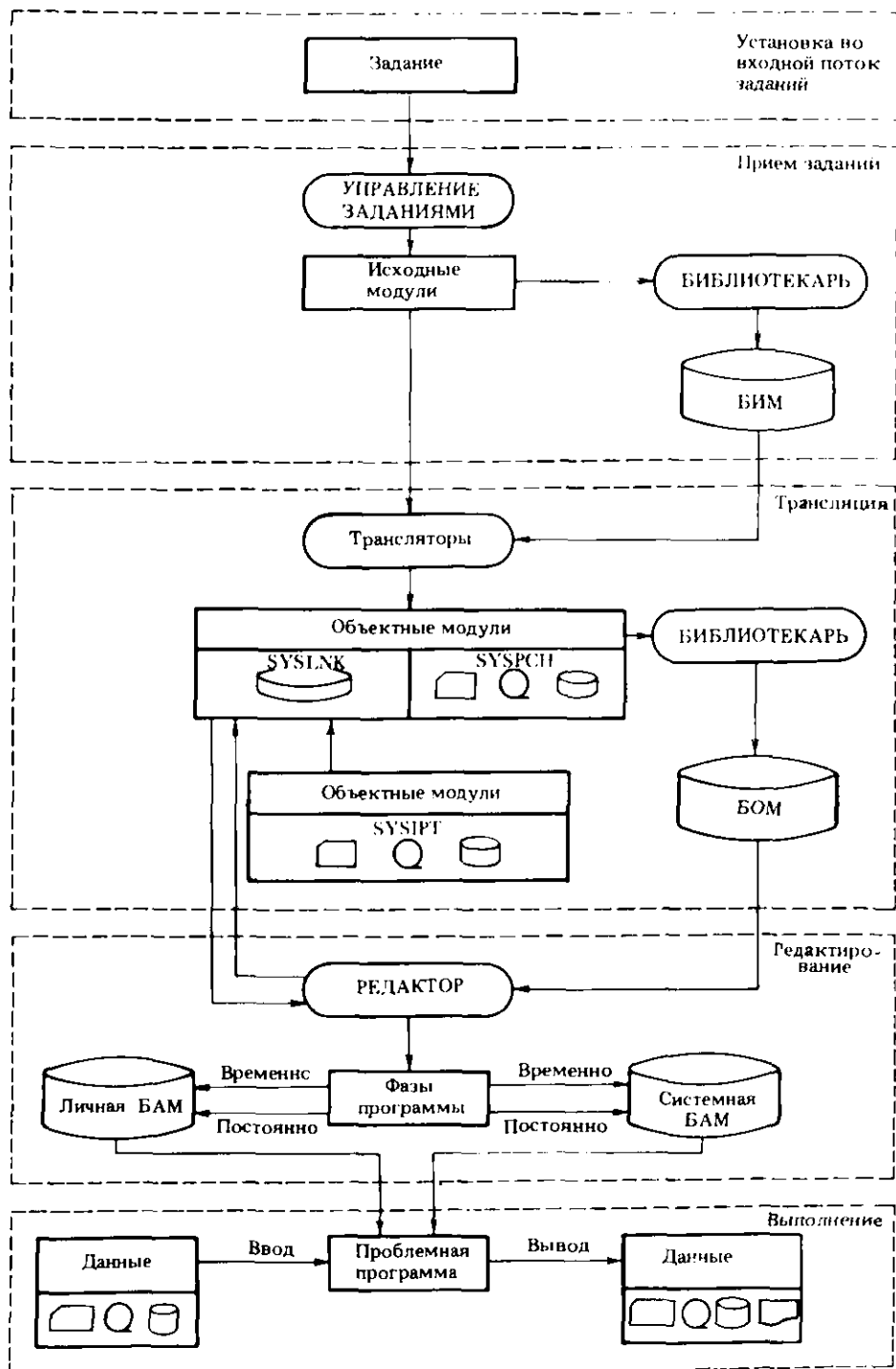


Рис. 17.

Приведем примеры нескольких вариантов заданий на трансляцию с необходимыми пояснениями.

Пример 1. Составить задание на трансляцию программы RASNAGA1, написанной на ФОРТРАНе IV, с выдачей на SYSLST исходного текста программы, сообщений об ошибках и информации о распределении памяти (в дальнейших примерах, если не оговорено противное, предполагается, что устройства SYSRDR и SYSIPT совмещены).

Задание операционной системе, обеспечивающее выполнение этапов обработки программы RASNAGA1, указанных в условии примера, имеет вид:

```
// JOB ОТЛАДКА
// ASSGN SYSLST, X'00F'
// OPTION LIST, NOLISTX, NODECK, NOLOG, NODUMP
// EXEC FFORTRAN
      (Файл с текстом программы RASNAGA1)
/*
/ &
```

П о я с н е н и я. 1. FFORTRAN – имя транслятора с ФОРТРАНа IV, использованное при каталогизации в системную библиотеку абсолютных модулей.

2. Вывод указанной в задании информации на печать обеспечивается режимным требованием LIST в операторе OPTION.

3. Режимные требования NOLISTX, NODECK, NOLOG, NODUMP отменяют соответственно LISTX, DECK, LOG, DUMP, если эти режимные требования действовали к началу данного задания.

Назначение последних следующее: LISTX сообщает, что на SYSLST должна быть выведена распечатка объектных модулей транслируемой программы; DECK требует вывода карт (или их образов) объектных модулей на SYSPCH; LOG вызывает вывод на SYSLST всех действующих операторов и директив программы УПРАВЛЕНИЕ ЗАДАНИЯМИ; DUMP в случае сбоя (ненормального завершения) работающей на данном шаге задания программы (в рассматриваемом примере это программа FFORTRAN) вызывает вывод на SYSLST содержимого основной памяти и регистров арифметическо-логического устройства (АЛУ) ЭВМ в шестнадцатичном коде.

П р и м е ч а н и я. 1. При действии режима NOLIST вывод на SYSLST сообщений об ошибках остается в силе.

2. В последующих примерах предполагаются стандартными режимы LIST, NOLISTX, NODECK, NOLOG, NODUMP и назначение физического устройства ввода-вывода X'00F' логическому устройству SYSLST.

Пример 2. Составить задание на трансляцию RASNAGA1 и модуля GRAFIKYX, написанного на АССЕМБЛЕРе, с печатью исходных и объектных модулей и выводом последних на SYSPCH.

Задание операционной системе, обеспечивающее выполнение этапов обработки программы RASNAGA1 и модуля GRAFIKYX, указанных в условии примера, имеет вид:

```
// JOB ОБЪЕКТ
// ASSGN SYSPCH, X'00D'
// OPTION LISTX, DECK
// EXEC FFORTRAN
      (Файл с текстом программы RASNAGA1)
/*
// EXEC ASSEMBLF
      (Файл с текстом модуля GRAFIKYX)
/*
/ &
```

П о я с н е н и я: ASSEMBLF – имя транслятора с АССЕМБЛЕРа.

Пример 3. При сохранении остальных условий предыдущего примера не выводятся на SYSLST исходный и объектный модули GRAFIKYX.

Задание операционной системе, обеспечивающее обработку программы RASNAGA1 и модуля GRAFIKYX, указанных в условии примера, имеет вид:

```
// JOB ОБ'ЕКТ1
// ASSIGN SYSPCH, X'00D'
// OPTION LISTX, DECK
// EXEC FFORTRAN
    (Файл с текстом программы RASNAGAI)
/*
// OPTION NOLIST, NOLISTX
// EXEC ASSEMBL.F
    (Файл с текстом модуля GRAFIKYX)
/*
/&
```

Каталогизация в библиотеки исходных модулей

Каталогизация исходных модулей в соответствующие личные или системную библиотеки осуществляется системной программой, каталогизированной в системной библиотеке абсолютных модулей (БАМ) под именем MAINT и являющейся частью БИБЛИОТЕКАРЯ. После исполнения программы MAINT всегда выводится на SYSLSY таблица состояния той библиотеки, в которую производилась каталогизация модуля. В этой таблице содержится информация об областях памяти на дисках, занимаемых библиотекой и ее оглавлением, а также о заполненности библиотеки и оглавления.

Для каталогизации в библиотеки исходных модулей (БИМ) модуль, независимо от языка программирования, на котором он написан, должен быть дополнен двумя картами — в начале и в конце модуля, содержащими специальный оператор BKEND (в любых колонках, кроме 77–80). После этого модуль может быть воспринят программой MAINT как подготовленная к каталогизации книга. Необходимую для программы MAINT информацию о подбиблиотеке системной или личной библиотек исходных модулей, в которую должна быть помещена книга, и об имени, под которым ее следует каталогизировать, сообщает управляющий оператор БИБЛИОТЕКАРЯ CATALS. Этот оператор, как и другие управляющие операторы БИБЛИОТЕКАРЯ, используемые программой MAINT, во входном потоке заданий должен размещаться вслед за управляющим оператором программы УПРАВЛЕНИЕ ЗАДАНИЯМИ

// EXEC MAINT

Оператор CATALS имеет следующий формат:

CATALS имя подбиблиотеки. имя книги

Здесь "имя подбиблиотеки" — параметр, значение которого должно быть имя подбиблиотеки соответствующего языка программирования, принятое при создании библиотеки. В качестве имени подбиблиотеки может использоваться любой одиночный цифровой или алфавитный символ. Параметру "имя книги" задается значение имени, под которым пользователь желает каталогизировать книгу. Имя книги может содержать от 1 до 8 алфавитно-цифровых символов ФОРТРАНа, первый из которых должен быть буквой. Между параметрами обязательна точка в качестве разделителя. Необязательные параметры оператора в приведенном формате не показаны.

При использовании личной библиотеки исходных модулей в задание должен обязательно помещаться управляющий оператор программы УПРАВЛЕНИЕ ЗАДАНИЯМИ, который сообщает операционной системе идентифицирующую информацию для отыскания на физическом устройстве ввода-выво-

да, соответствующем SYSSLB, файла, содержащего личную библиотеку исходных модулей. В простейшем случае, когда на пакете магнитных дисков размещено не более одной личной библиотеки исходных модулей, указанный оператор имеет вид

```
// DLBL IJSYSSL
```

Пример 4. Составить задание на каталогизацию книги RASNAGA1 в системную библиотеку исходных модулей.

Задание операционной системе, обеспечивающее выполнение обработки программы RASNAGA1, указанной в условии примера, имеет вид:

```
// JOB КАТАЛОГ1
// ASSGN SYSRES,X'190'
// EXEC MAINT
  CATALS 3. RASNAGA 1
  (Файл с текстом книги RASNAGA1)
/*
/ &
```

П о я с н е н и е. Символ "3" в операторе CATALS означает имя подбиблиотеки ФОРТРАНа IV, принятое по предположению в системной библиотеке исходных модулей.

Пример 5. Составить задание на каталогизацию книги RASNAGA1 в личную библиотеку исходных модулей.

Задание операционной системе, обеспечивающее выполнение обработки программы RASNAGA1, указанной в условии примера, имеет вид:

```
// JOB КАТАЛОГ2
// ASSGN SYSSLB,X'191'
// DLBL IJSYSSL
// EXEC MAINT
  CATALS 1. RASNAGA 1
  (Файл с текстом книги RASNAGA1)
/*
/ &
```

Пример 6. Составить задание на каталогизацию книг GRAFIKYX и RASNAGA1 соответственно в системную и личную библиотеки исходных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB КАТАЛОГ3
// ASSGN SYSRES,X'190'
// EXEC MAINT
  CATALS 1.GRAFIKYX
  (Файл с текстом книги GRAFIKYX)
/*
// ASSGN SYSSLB,X'191'
// EXEC MAINT
  CATALS 1.RASNAGA 1
  (Файл с текстом книги RASNAGA1)
/*
/ &
```

Каталогизация в библиотеки объектных модулей

Для каталогизации объектных модулей в системную или личную библиотеку используется, как уже упоминалось, программа MAINT. Необходимую информацию пользователь сообщает с помощью управляющего оператора БИБЛИОТЕКАРЯ CATALR, который имеет формат

CATALR имя модуля

(необязательные параметры не показаны). Параметру "имя модуля" задается в качестве значения имя, под которым должен быть каталогизирован модуль,

установленный на SYSIPT (или SYSIN) на данном шаге задания (от 1 до 8 символов ФОРТРАНА, первый из которых не должен быть *).

При использовании личной библиотеки объектных модулей (БОМ) задание должно содержать управляющий оператор программы УПРАВЛЕНИЕ ЗАДАНИЯМИ вида

```
// DLBL IISYSRL
```

Он необходим для той же цели, что и оператор DLBL, используемый при каталогизации в личную библиотеку исходных модулей.

Пример 7. Составить задание на каталогизацию объектного модуля RASNAGA1 в личную библиотеку объектных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB КАТАЛОГ4
// ASSGN SYSRLB,X'191'
// DLBL IISYSRL
// EXEC MAINT
  CATALR RASNAGA1
  (Файл с текстом объектных модулей программы)
/*
/&
```

Если модуль, подлежащий каталогизации в библиотеку объектных модулей, еще не оттранслирован, то необходимо при его трансляции задать требование DECK в операторе OPTION и назначить НМЛ логическому устройству SYSPCH. По окончании шага задания, на котором выполнялась трансляция, необходимо записать на НМЛ ленточную марку, служащую для указания конца файла на ленте, назначить указанный НМЛ устройству SYSIPT, перемотать ленту к началу файла и провести требуемую каталогизацию.

Для записи ленточной марки и перемотки ленты можно использовать управляющий оператор MTC программы УПРАВЛЕНИЕ ЗАДАНИЯМИ. Он имеет формат

```
// MTC код служебной операции с лентой, адрес НМЛ
```

Параметру "код служебной операции с лентой" для записи ленточной марки присваивается значение WTM, для перемотки ленты к началу файла — значение BSF; параметру "адрес НМЛ" присваивается либо его адрес как физического устройства ввода-вывода, либо последнее назначенное этому НМЛ логическое устройство. Необязательные параметры оператора MTC в приведенном формате не показаны.

Пример 8. Составить задание на трансляцию и каталогизацию в системную библиотеку объектных модулей программы RASNAGA1.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB ТРАНСБОБ
//ASSGN SYSPCH,X'280'
// OPTION DECK
// EXEC FFORTRAN
  (Файл с текстом исходных модулей программы RASNAGA1)
/*
// MTC WTM,SYSPCH
// ASSGN SYSIPT,X'280'
// MTC BSF, X'280'
// EXEC MAINT
  CATALR RASNAGA1
/*
/&
```

Оператор PHASE. Управляющие операторы PHASE служат для сообщения РЕДАКТОРУ сведений о желаемой фазовой структуре программы и областях памяти, в которые будут загружаться фазы для исполнения.

Расположение текстов фаз, составленных из объектных модулей, на SYSIPT должно соответствовать расположению относящихся к этим фазам операторов PHASE на SYSRDR, а в случае совмещения SYSIPT и SYSRDR в одном физическом устройстве текст фазы должен следовать за соответствующим оператором PHASE. Если в данном задании для некоторых фаз должно быть произведено не только редактирование, но и предшествующая ему трансляция, эти же правила распространяются и на расположение фаз, составленных из исходных модулей. Тексты отдельных фаз могут находиться также в системных или личных библиотеках объектных или исходных модулей.

Формат оператора PHASE следующий:

PHASE имя фазы, адрес загрузки

(здесь и далее необязательные параметры не показаны). Параметру "имя фазы" задается в качестве значения имя, под которым фаза должна каталогизироваться в системную или личную библиотеку абсолютных модулей. Первые части символа имен всех фаз, относящихся к данной программе, должны совпадать. Значение параметра "адрес загрузки" определяет истинный адрес, начиная с которого будет размещаться фаза при вызове в основную память.

Наиболее употребительны следующие способы задания адреса загрузки:

1) для корневой фазы, т.е. такой фазы, которая будет постоянно находиться в основной памяти при работе программы и производить вызов из библиотеки других фаз, задается значение ROOT, указывающее, что фаза должна загружаться, начиная с первого свободного адреса, не занятого системными программами;

2) для других фаз адрес загрузки задается либо в виде

{ $\pm$ смещение}

означающем, что адрес будет определяться как первый свободный адрес за предыдущей фазой, измененный на величину смещения (если последнее задано), которое может содержать от 1 до 8 десятичных цифр, либо в виде

имя фазы { $\pm$ смещение}

где параметру "имя фазы" в качестве значения задается имя одной из фаз данной программы, для которой адрес загрузки задан в другом операторе PHASE, а смещение задается по отношению к адресу загрузки указанной фазы.

При отсутствии в задании операторов PHASE редактируемая программа рассматривается РЕДАКТОРОМ как фиктивная фаза и должна исполняться непосредственно после редактирования, для чего необходимо использовать оператор EXEC без параметра.

Оператор INCLUDE. Вся входная информация для РЕДАКТОРА должна находиться на SYSLNK. Управляющие операторы РЕДАКТОРА переносятся с SYSRDR на SYSLNK программой УПРАВЛЕНИЕ ЗАДАНИЯМИ. Исходные модули после трансляции в объектные помещаются на SYSLNK трансляторами. Для записи на SYSLNK объектных модулей, находящихся на SYSIPT, а также первого (или единственного) модуля редактируемой программы из

библиотеки объектных модулей необходим специальный управляющий оператор. Эту функцию выполняет оператор INCLUDE.

Кроме того, оператор INCLUDE можно использовать для указания РЕДАКТОРУ имен тех модулей из системной и личной библиотек объектных модулей, которые будут включены в фазы программы в соответствии с имеющимися в исходных модулях внешними именами. Это целесообразно в тех случаях, когда операторы INCLUDE позволяют РЕДАКТОРУ построить более эффективную программу (с точки зрения занимаемой ею памяти) по сравнению с вариантом программы, где подключение требуемых модулей производится непосредственно по их именам в тексте исходных модулей.

Каждый оператор INCLUDE относится к той фазе, которая определяется предшествующим оператором PHASE.

Формат оператора следующий:

INCLUDE [имя модуля]

Параметру "имя модуля" в качестве значения задается имя того модуля, который необходимо включить в данную фазу. При отсутствии параметра по оператору INCLUDE программа УПРАВЛЕНИЕ ЗАДАНИЯМИ переписывает на SYSLNK объектный модуль, находящийся на SYSIPT (при этом сам оператор INCLUDE в отличие от других операторов РЕДАКТОРА на SYSLNK не записывается).

Если обращение к некоторому модулю, находящемуся в одной из библиотек объектных модулей, встречается в нескольких фазах редактируемой программы, то оператор INCLUDE с именем этого модуля целесообразно расположить в задании так, чтобы он включался в корневую фазу. Тогда во всех фазах, обращающихся к данному модулю, РЕДАКТОР создает необходимые ссылки к нему, не включая его текст в эти фазы.

При создании фаз для автоматического подключения необходимых программ из системной и личной библиотек объектных модулей задается режим AUTOLINK. Если задан режим NOAUTO, то подключение этих программ - дело программиста.

Источники входной информации РЕДАКТОРА показаны на рис. 18.

Составление заданий на редактирование, исполнение и каталогизацию в библиотеки абсолютных модулей

Системная программа РЕДАКТОР каталогизирована в системную библиотеку абсолютных модулей под именем LNKEDT. Управляющие операторы РЕДАКТОРА должны предшествовать оператору //EXEC LNKEDT.

При использовании личной библиотеки абсолютных модулей задание должно содержать директиву (а не оператор!) назначения физического устройства ввода-вывода логическому устройству SYSCLB и оператор программы УПРАВЛЕНИЕ ЗАДАНИЯМИ

//DLBL USYSCLB

Приведем примеры заданий, включающих редактирование объектных модулей.

Пример 9. Составить задание на редактирование объектных модулей RASNAGA1, размещенных на SYSIN, и исполнение в виде однофазовой программы.

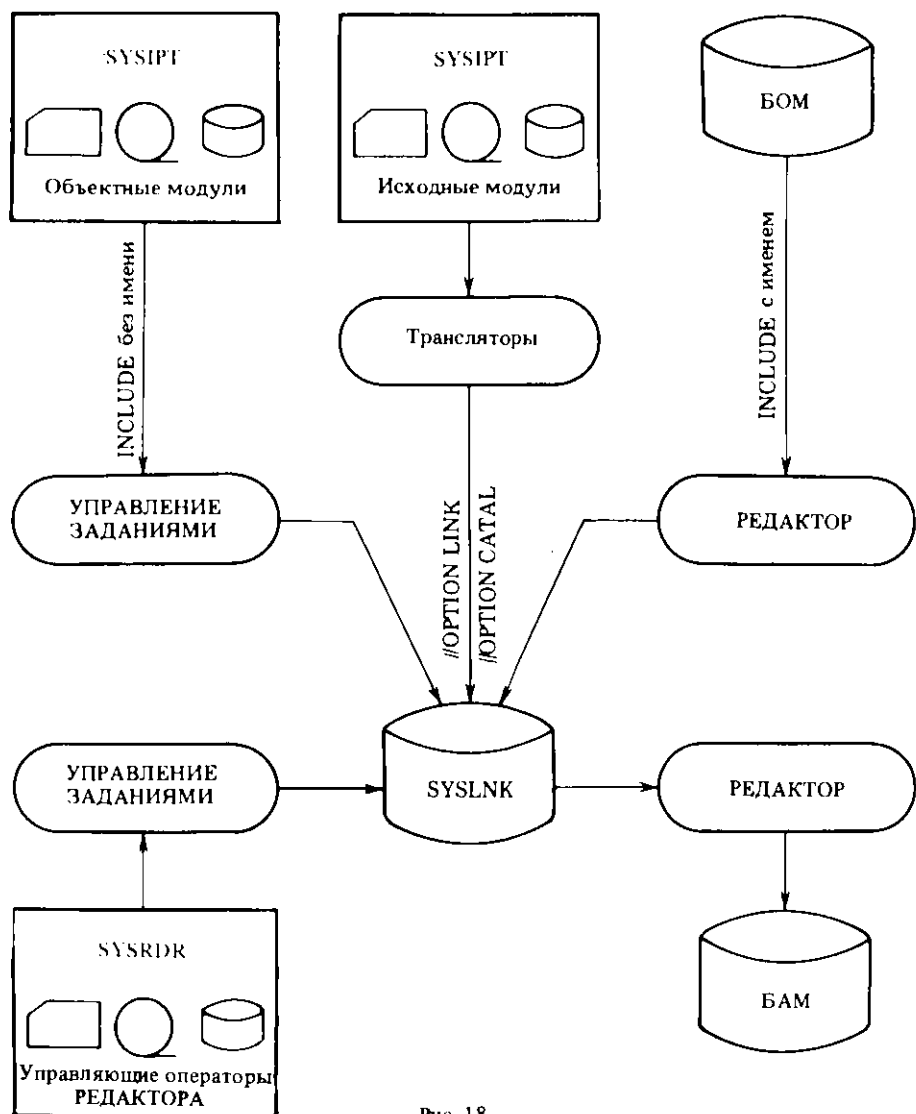


Рис. 18.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB РЕДИСП1
INCLUDE
  (Файл с текстом объектных модулей программы RASNAGA1)
/*
// EXEC LNKEDT
// EXEC
  (Файл с данными)
/*
/&
```

Пример 10. Условия такие же, как и в примере 9, но RASNAGA1 – исходный модуль.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB РЕДИСП2
// OPTION LINK
// EXEC FFORTRAN
(Файл с текстом исходных модулей программы RASNAGA1)
/*
// EXEC LNKEDT
// EXEC
(Файл с данными)
/*
/&
```

Пример 11. Условия такие же, как в примере 9, но программа RASNAGA1 находится в личной библиотеке объектных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB РЕДИСП3
// ASSGN SYSRLB,X'191'
// DLBL IFSYRL
INCLUDE RASNAGA1
// EXEC LNKEDT
// EXEC
(Файл с данными)
/*
/&
```

Пример 12. Условия такие же, как и в примере 11, но RASNAGA1 находится в системной библиотеке исходных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB РЕДИСП4
// ASSGN SYSRFS,X'190'
// ASSGN SYSPCH,X'280'
// EXEC SSERV
PUNCH 3.RASNAGA1
// MTC WTM,X'280'
// ASSGN SYSIPT,X'280'
// MTC BSF,X'280'
// OPTION LINK
// EXEC FFORTRAN
// EXEC LNKEDT
// EXEC
(Файл с данными)
/*
/&
```

Пример 13. Составить задание на редактирование и исполнение программы RASNAGA 2, размещенной на SYSIN в виде объектных модулей, если известно, что она должна состоять из корневой фазы RASNAGA2 и фазы RASNDNPR, частично перекрывающей в основной памяти фазу RASNAGA2.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB РЕДИСП5
PHASE RASNAGA2, ROOT
INCLUDE
(Файл с текстом объектного модуля RASNAGA2)
```

```

/*
  PHASE RASNDNPR, *-1000
  INCLUDE
    (Файл с текстом объектного модуля RASNDNPR)
/*
  // OPTION CATAL
  // EXEC LNKEDT
  // EXEC RASNAGA2
    (Файл с данными)
/*
/&

```

Примечание. Для редактирования и исполнения в одном задании многофазовой программы необходим режим CATAL, а не LINK, как в случае фиктивной фазы.

Пример 14. Условия такие же, как и в примере 13, но RASNAGA2 — исходный модуль, и, кроме того, в фазу RASNDNPR необходимо включить модуль GRAFIKYX из личной библиотеки объектных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```

// JOB РЕДИСП6
// ASSGN SYSRLB,X'191'
// DLBL IJSYRL
// OPTION CATAL
  PHASE RASNAGA2, ROOT
// EXEC FFORTRAN
    (Файл с текстом исходного модуля RASNAGA2)
/*
  PHASE RASNDNPR, *-1000
  // EXEC FFORTRAN
    (Файл с текстом исходного модуля RASNDNPR)

/*
  INCLUDE GRAFIKYX
// EXEC LNKEDT
// EXEC RASNAGA2
    (Файл с данными)
/*
/&

```

Пример 15. Составить задание на редактирование и каталогизацию однофазовой программы RASNAGA1 с SYSIN в системную библиотеку абсолютных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```

// JOB РЕДКАТЛ1
// OPTION CATAL
  PHASE RASNAGA1, ROOT
  INCLUDE
    (Файл с текстом объектных модулей программы RASNAGA1)
/*
// EXEC LNKEDT
/&

```

Пример 16. Условия такие же, как и в примере 15, но каталогизацию произвести в личную библиотеку абсолютных модулей.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```

// JOB РЕДКАТЛ2
  ASSGN SYSCLB,X'191'
// DLBL IJSYSL
// OPTION CATAL
  PHASE RASNAGA1, ROOT

```

```

INCLUDE
(Файл с текстом объектных модулей программы RASNAGA1)
/*
// EXEC LINKEDT
// ASSGN SYSCLB,UA
/&

```

Распечатка исходных данных

Системная программа перезаписи КАРТЫ-ПЕЧАТЬ/КАРТЫ, осуществляющая распечатку, каталогизирована под именем CDPР. Программа имеет собственные управляющие операторы. Если карты пронумерованы в колонках 79—80, как и в лабораторной работе 1, то распечатку можно производить с помощью следующего задания:

```

// JOB РАСПЧАТКА
// ASSGN SYS004,X'00C'
// ASSGN SYS005,X'00F'
// UCPTD,FF,A=(80,80),B=(120),Q=(79,2)
// HI ..... РАСПЕ-
ЧАТКА_КАРТ_УПРАВЛЕНИЯ_И_ДАННЫХ
// EXEC CDPР
// END
(Файл с данными)
/*
/& ..... РАСПЕЧАТКА ВЫПОЛНЕНА

```

2. ПРОГРАММЫ ОПТИМАЛЬНОГО РАСПРЕДЕЛЕНИЯ АКТИВНЫХ МОЩНОСТЕЙ МЕЖДУ АГРЕГАТАМИ ЭЛЕКТРОСТАНЦИИ

В данной лабораторной работе используются программы RASNAGA1, RASNAGA2, предназначенные для решения задачи оптимизации (1)–(3).

Каждая из программ RASNAGA1 и RASNAGA2 состоит из основной программы и подпрограммы DYNPRG. Программа RASNAGA1 работает как однофазовая. Это позволяет ускорить выполнение программы благодаря тому, что при ее работе не затрачивается время на вызов в основную память некорневых фаз. Программа RASNAGA2 является двухфазовой. Корневая фаза RASNAGA2 вызывает фазу RASNDNPR, которая частично перекрывает корневую фазу, благодаря чему экономится память, занимаемая программой в основной памяти. Подробная информация о программе содержится в лабораторной работе 1. Здесь же приводятся лишь сведения, необходимые для составления различных вариантов заданий для ДОС/ЕС–2.1 при работе с программами RASNAGA1, RASNAGA2.

3. ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

При подготовке к выполнению лабораторной работы необходимо:

1) изучить раздел 1 данной лабораторной работы, обратив особое внимание на форматы управляющих операторов и методику составления различных вариантов заданий для ДОС/ЕС–2.1;

- 2) изучить раздел 2 данной лабораторной работы;
- 3) получить у преподавателя один из вариантов задания на выполнение данной лабораторной работы;
- 4) составить задание для ДОС/ЕС–2.1 в соответствии с полученным вариантом для сдачи на перфорацию.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать в себя следующие разделы:

- 1) цель работы;
- 2) предварительно подготовленный студентами и исправленный преподавателем вариант задания для ДОС/ЕС–2.1;
- 3) описание назначения и формата всех управляющих операторов, использованных в задании;
- 4) распечатку выходной информации ДОС/ЕС–2.1 по заданию;
- 5) схему возможных этапов обработки заданий операционной системой ДОС/ЕС;
- 6) схему возможных источников входной информации для РЕДАКТОРА.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что представляет собой операционная система ЭВМ и каково ее назначение?
2. Укажите основные функции операционной системы ДОС/ЕС–2.1.
3. Каково назначение системных программ СУПЕРВИЗОР и УПРАВЛЕНИЕ ЗАДАНИЯМИ?
4. С какой целью в системе ДОС/ЕС-2.1 принято использовать логические устройства ввода-вывода? Перечислите эти устройства и назовите их функции.
5. Каковы формат и функции основных управляющих операторов программы УПРАВЛЕНИЕ ЗАДАНИЯМИ?
6. Объясните сущность различных этапов обработки задания для ДОС/ЕС–2.1, показанных на рис. 17.
7. Для чего служат управляющие операторы операционной системы?
8. Для чего осуществляется каталогизация программы в библиотеку абсолютных модулей?
9. Какие управляющие операторы и в каком порядке должны использоваться при составлении различных вариантов заданий на трансляцию исходных модулей?
10. Укажите виды библиотек, допускаемых операционной системой ДОС/ЕС–2.1, и охарактеризуйте способы составления заданий на каталогизацию программ в эти библиотеки.
11. Каков наиболее предпочтительный, на ваш взгляд, способ хранения и ввода в основную память ЭВМ программ в условиях АСУ?
12. Каковы форматы и функции основных управляющих операторов РЕДАКТОРА?
13. Объясните различие между понятиями модуль и фаза.

14. Охарактеризуйте способы получения входной информации РЕДАКТО-
РОМ при формировании им текста фаз.
15. Для чего создаются многофазовые программы?
16. В чем состоят особенности корневой фазы?
17. Как происходит вызов в основную память некорневых фаз?
18. Охарактеризуйте способы составления заданий для ДОС/ЕС—2.1 на редак-
тирование и выполнение программ.
19. Каковы способы контроля правильности исходных данных?

6. ВАРИАНТЫ ИСХОДНЫХ ДАННЫХ

Составить задания для операционной системы ДОС/ЕС—2.1 на этапы обработки проблемных программ, указанные в соответствующем варианте исходных данных.

Вариант 1. Трансляция, редактирование и каталогизация в личную библиотеку абсолютных модулей программы RASNAGA1, текст которой находится на SYSIPT.

Вариант 2. Трансляция, редактирование и каталогизация в системную библиотеку абсолютных модулей программы RASNAGA2, текст которой находится на SYSIPT.

Вариант 3. Трансляция, редактирование без перекрытия фаз и исполнение программы RASNAGA2, текст которой находится на SYSIPT.

Вариант 4. Редактирование, каталогизация в личную библиотеку абсолютных модулей и исполнение программы RASNAGA1, текст которой находится в личной библиотеке объектных модулей.

Вариант 5. Редактирование, каталогизация в системную библиотеку абсолютных модулей и исполнение программы RASNAGA2, текст которой находится в личной библиотеке объектных модулей.

Вариант 6. Редактирование без перекрытия фаз и исполнение программы RASNAGA2, если фаза RASNAGA2 находится в системной библиотеке объектных модулей, а фаза RASNDNPR — в личной библиотеке объектных модулей.

Вариант 7. Редактирование без перекрытия фаз и каталогизация в личную библиотеку абсолютных модулей программы RASNAGA2, если фаза RASNAGA2 находится в личной библиотеке объектных модулей, а фаза RASNDNPR — в системной библиотеке объектных модулей.

Вариант 8. Трансляция, редактирование, каталогизация в системную библиотеку абсолютных модулей и исполнение программы RASNAGA1, текст которой находится на SYSIPT.

ОРГАНИЗАЦИЯ КОРРЕКТИРУЕМОГО ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК ЭЛЕКТРОСТАНЦИЙ ДЛЯ ИСПОЛЬЗОВАНИЯ ПРИ ОПЕРАТИВНОЙ ОПТИМИЗАЦИИ РЕЖИМА ЭНЕРГОСИСТЕМЫ

Ц е л ь р а б о т ы: ознакомиться с декомпозиционным подходом к решению задач оптимизации большой размерности на примере решения задачи оптимального распределения активных мощностей между электрическими станциями энергосистемы методом динамического программирования; получить навыки в организации хранения больших объемов условно-постоянной информации об энергосистеме в файлах прямого доступа и в эксплуатации проблемных программ, допускающих несколько возможных режимов работы.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Декомпозиция задачи оптимального распределения активных мощностей в энергосистеме на подзадачи внутри- и межстанционной оптимизации

В лабораторной работе 1 по данному курсу рассматривалась задача определения оптимальных активных мощностей агрегатов электростанции, если задана ее суммарная активная мощность. В действительности при управлении режимом энергосистемы по активной мощности исходной режимной информацией для принятия решения является прогноз активных мощностей потребления в узлах основной сети энергосистемы, а мощности заранее неизвестны, и требуется определить оптимальные значения мощностей одновременно для всех агрегатов энергосистемы.

Допустим, что потери активной мощности в основной сети в планируемом режиме мало влияют на величины искомым оптимальных мощностей агрегатов, а поэтому могут быть заранее спрогнозированы и включены в величину суммарной мощности потребления в энергосистеме. С учетом этого задача оптимального распределения активных мощностей между агрегатами энергосистемы по своей математической формулировке принципиально не отличается от задачи внутростанционной оптимизации (см. лабораторную работу 1):

$$\sum_{i=1}^m \sum_{j=1}^{n_i} \varphi_{ij}(P_{ij}) \rightarrow \min; \quad (10)$$

$$\sum_{i=1}^m \sum_{j=1}^{n_i} P_{ij} = P; \quad (11)$$

$$P_{ij\min} \leq P_{ij} \leq P_{ij\max}, j = 1, 2, \dots, n_i, i = 1, 2, \dots, m, \quad (12)$$

где m — количество электростанций в энергосистеме; n_i — количество агрегатой i -й станции; P_{ij} — мощность j -го агрегата i -й станции; $\varphi_{ij}(P_{ij})$ — его расходная

характеристика; P — заданная суммарная мощность потребления в энергосистеме.

Однако ввиду значительно большей размерности (количества независимых переменных) задачи (10) — (12) по сравнению с задачей внутристанционной оптимизации для обеспечения требуемой в условиях эксплуатации оперативности принятия решений необходимо найти специальные вычислительные методы, существенно снижающие затраты машинного времени на отыскание оптимальных значений переменных P_{ij} по сравнению с прямым применением МДП к решению задачи (10) — (11). Эффективным методом решения задач большой размерности является их декомпозиция, т.е. сведение их решения к решению ряда подзадач меньшей размерности. В данном случае такими подзадачами будут следующие.

1. Множество задач внутристанционной оптимизации, решаемых при построении расходных характеристик $\varphi_i(P_i)$, $i = 1, 2, \dots, m$, электростанций, где P_i — текущее значение суммарной активной мощности i -й станции, изменяемое с некоторым шагом ΔP_i на интервале

$$\sum_{j=1}^{n_i} P_{ij\min} \leq P_i \leq \sum_{j=1}^{n_i} P_{ij\max}. \quad (13)$$

При этом каждому значению P_i соответствует отдельная задача оптимизации, состоящая в оптимальном распределении суммарной мощности P_i между агрегатами i -й станции.

2. Задача межстанционной оптимизации, состоящая в отыскании оптимальных значений P_i , $i = 1, 2, \dots, m$, суммарных мощностей станций при заданной мощности потребления P в энергосистеме и известных расходных характеристиках станций.

Таким образом, построение расходной характеристики i -й станции сводится к решению ряда задач оптимизации вида:

$$\sum_{j=1}^{n_i} \varphi_{ij}(P_{ij}) \rightarrow \min; \quad (14)$$

$$\sum_{j=1}^{n_i} P_{ij} = P_i; \quad (15)$$

$$P_{ij\min} \leq P_{ij} \leq P_{ij\max}, \quad j = 1, 2, \dots, n_i, \quad (16)$$

где очередная конкретная задача вида (14) — (16) соответствует выбору очередного значения P_i из интервала (13).

Каждая из задач вида (14) — (16) решается методом динамического программирования. При этом с вычислительной точки зрения весьма важен тот факт, что наиболее трудоемкую часть алгоритма решения задач (14) — (16) для данной i -й станции (прямой ход динамического программирования) достаточно выполнить только один раз, так как одна и та же функция Беллмана используется при любом значении P_i .

Задача межстанционной оптимизации

$$\sum_{i=1}^m \varphi_i(P_i) \rightarrow \min; \quad (17)$$

$$\sum_{i=1}^m P_i = P; \quad (18)$$

$$P_{i \min} \leq P_i \leq P_{i \max}, \quad i = 1, 2, \dots, m; \quad (19)$$

$$P_{i \min} = \sum_{j=1}^{n_i} P_{ij \min}, \quad P_{i \max} = \sum_{j=1}^{n_i} P_{ij \max}$$

также решается методом динамического программирования.

Таким образом, как задача вида (14) – (16), так и задача (17) – (19) могут быть решены с помощью подпрограммы DYNPRG, используемой в программах RASNAGA1, RASNAGA2 оптимального распределения мощностей между агрегатами станции (см. лабораторную работу 1), если включить эту подпрограмму в программу оптимального распределения активных мощностей между агрегатами энергосистемы при редактировании последней (см. лабораторную работу 2).

В процессе функционирования энергосистемы составы агрегатов, включенных в работу на электростанциях, неоднократно изменяются даже на протяжении одних суток. Поэтому при управлении режимом энергосистемы целесообразно хранить во внешней памяти ЭВМ наборы заблаговременно построенных расходных характеристик всех электростанций, соответствующие различным составам включенного в работу оборудования, ранее встречавшимся в процессе эксплуатации данной задачи. Совокупность таких наборов характеристик образует базу условно-постоянной информации (УПИ) для задачи оптимизации режима энергосистемы по активной мощности. При наличии такой базы для отыскания оптимального режима в большинстве случаев достаточно будет решить только межстанционную задачу оптимизации (17) – (19), что существенно повышает оперативность принятия диспетчерских решений. Для тех случаев, когда планируемые в текущем режиме составы оборудования на электростанциях не представлены расходными характеристиками станций в базе УПИ, в составе комплекса программы данной задачи АСУ должна быть предусмотрена программа, осуществляющая построение указанных характеристик и пополнение ими имеющейся базы УПИ.

Общие сведения об организации хранения данных в файлах на внешних носителях информации для ЕС ЭВМ и методах обработки файлов в операционной системе ДОС/ЕС

Задачи управления энергосистемами, как и любыми народнохозяйственными объектами, в большинстве случаев требуют хранения и обработки больших объемов данных – *нормативно-справочной информации*, содержащей характеристики оборудования энергосистемы, установленные нормативы различного типа и т.д.; *условно-постоянной информации*, отражающей относительно медленно меняющуюся структуру энергосистемы и ее подразделений,

плановые показатели для принятых календарных периодов планирования, ретроспективные данные о предыстории изменения во времени различных производственных процессов; *переменной информации*, порождаемой текущим процессом функционирования энергосистемы либо возникающей как промежуточные данные при решении сложных задач большой размерности.

Соответствующие группы данных должны храниться на внешних носителях информации, принятых в ЕС ЭВМ, в форме, удобной для обмена с основной памятью ЭВМ, для программной обработки данных и (или) получения справок по запросам пользователя в виде тех или иных документов. Группа данных, объединяемых общностью своего целевого назначения и хранящихся на внешних носителях информации для ЕС ЭВМ, называется *файлом*. С точки зрения пользователя, элементами файла являются так называемые логические записи, которые представляют собой подмножество данных (чисел, слов или предложений), объединяемых по смысловому признаку и располагаемых последовательно на физических носителях файла. При обработке файлов естественными элементами файла являются так называемые физические записи, максимальный размер которых определяется физической конструкцией соответствующих носителей информации. Например, для файлов на перфокартах максимальный размер физической записи равен 80 байтам в соответствии с количеством позиций на перфокарте. Во многих случаях языки программирования операционной системы ДОС/ЕС позволяют оперировать логическими записями, размер которых может превышать предельный размер физических записей для соответствующего устройства ввода-вывода, а преобразование логических записей в физические, как и широкий набор других сервисных функций, резко упрощающих работу с файлами, выполняется программами СИСТЕМЫ УПРАВЛЕНИЯ ВВОДОМ-ВЫВОДОМ (СУВВ), входящими в состав УПРАВЛЯЮЩЕЙ ПРОГРАММЫ.

В зависимости от характера информации, помещаемой в файл, и от характера задачи, в которой этот файл используется, могут возникнуть две ситуации: 1) требуется однородная обработка больших "порций" информации последовательно с начала файла, содержащих значительные количества записей, которые также расположены в файле последовательно; 2) требуется обрабатывать записи, расположенные в произвольной области файла и характеризующиеся некоторым специфическим смысловым признаком.

Для обеспечения высокой скорости обмена информацией между основной и внешней памятью в обоих указанных случаях операционная система ДОС/ЕС предусматривает соответственно возможность создания файлов двух типов: 1) с последовательным доступом, которые позволяют обрабатывать записи только последовательно друг за другом, начиная с первой записи файла; 2) с прямым доступом, допускающие возможность обращения к произвольной записи.

Конструкция устройств памяти на магнитных дисках предоставляет возможность организации файлов обоих указанных типов. Все же другие внешние носители информации позволяют создавать только файлы с последовательным доступом. В программе POSHARST, используемой в данной работе, осуществляется организация файла на магнитных дисках с прямым доступом средствами языка ФОРТРАН IV.

Вследствие специфической конструкции устройств памяти на магнитных

дисках организация любых файлов на них позволяет программисту задать желаемый максимальный размер физической записи в пределах одной дорожки диска, что реализуется пользователем с помощью специальной системной программы ОЧИСТИТЬ ДИСК. Благодаря этому можно разместить логическую запись практически любой длины в пределах одной физической записи, что позволяет при программировании не разграничивать понятия физических и логических записей.

При программировании на языке ФОРТРАН IV необходимо иметь в виду различие между системными логическими устройствами и логическими устройствами программиста. В операторах ФОРТРАНа IV именам логических устройств взаимно однозначно соответствуют порядковые номера файлов, как показано в табл. 13.

Таблица 13

Имя логического устройства	SYSIPT	SYSIN	SYSPPH	SYSLST
Порядковый номер в операторах ФОРТРАНа IV	1	1	2	3
Имя логического устройства	SYSØØ1	SYSØØ2	SYSØØ3	SYSØØ4
Порядковый номер в операторах ФОРТРАНа IV	4	5	6	7
Имя логического устройства	SYSØØ5	SYSØØ6	SYSØØ7	SYSØØ8
Порядковый номер в операторах ФОРТРАНа IV	8	9	10	11
Имя логического устройства	SYSØØ9	SYSØ1Ø	SYSØ11	SYSLOG
Порядковый номер в операторах ФОРТРАНа IV	12	13	14	15

Формат операторов, обеспечивающих обработку файлов прямого доступа

Оператор программы УПРАВЛЕНИЕ ЗАДАНИЯМИ DLBL

Назначение этого оператора — сообщить операционной системе информацию, идентифицирующую файл, обрабатываемый в программе, которая должна выполняться на данном шаге задания или на последующих шагах.

Формат оператора:

// DLBL имя файла, [идентификатор файла], [срок сохранности], [код организации данных]

П р и м е ч а н и е. Приведенная запись формата более чем в одной строке текста, как и запись формата оператора EXTENT, не означает возможности использования карт продолжения при перфорации оператора, что недопустимо, как и для ранее рассмотренных управляющих операторов.

Параметр "имя файла" определяет логическое устройство ввода-вывода, предусматриваемое пользователем для размещения файла. По условиям работы системных программ СУВВ и трансляторов с языков программирования ДОС/ЕС значение этого параметра для выбранного логического устройства определяется в соответствии с табл. 14.

Таблица 14

Имя логического устройства	SYSIPT	SYSIN	SYSpch	SYSLST	SYSØØ1	SYSØØ2
Имя файла	IJSYSIN	IJSYSIN	IJSYSPH	IJSYSLs	IJSYSØ1	IJSYSØ2
Имя логического устройства	SYSØØ3	SYSØØ4	SYSØØ5	SYSØØ6	SYSØØ7	SYSØØ8
Имя файла	IJSYSØ3	IJSYSØ4	IJSYSØ5	IJSYSØ6	IJSYSØ7	IJSYSØ8
Имя логического устройства	SYSØØ9	SYSØ1Ø	SYSØ11	SYSsLB	SYSRLB	SYSCLB
Имя файла	IJSYSØ9	IJSYS1Ø	IJSYS11	IJSYSSL	IJSYSRL	IJSYSCL

Необязательный параметр "идентификатор файла" служит для указания пользователем операционной системе идентификатора обрабатываемого файла. Значением его может служить любой набор алфавитно-цифровых символов в количестве до 44, не считая апострофов. Если же этот параметр опущен, операционная система присваивает ему стандартное значение, совпадающее с именем файла, взятым в апострофы.

Значение параметра "срок сохранности" может задаваться либо в виде длительности периода сохранности файла, представленной числом, содержащим от одной до четырех десятичных цифр, либо в виде даты, по истечении которой не гарантируется сохранность файла, причем дата записывается в форме гг/ддд, где гг -- две последние цифры года окончания срока сохранности; ддд -- порядковый номер дня в указанном году. Если параметр опущен, операционная система принимает для файла стандартный срок сохранности, равный семи дням. Этому случаю эквивалентно также задание срока в виде ØØ/ØØØ.

Оператор программы УПРАВЛЕНИЕ ЗАДАНИЯМИ EXTENT

Назначение оператора -- сообщить операционной системе информацию об участке, занимаемом файлом в пакете магнитных дисков. Если файл используется в подлежащей исполнению программе как вводный, то операционная система уже обладает указанной информацией, так как она была сообщена в соответствующем операторе EXTENT при создании файла и учтена при формировании меток файла. Поэтому для вводных файлов оператор EXTENT не обязателен. Если же для таких файлов оператор EXTENT подготавливается пользователем, то любой параметр является необязательным. В случае выводного файла формат оператора следующий:

// EXTENT [имя логического устройства], [регистрационный номер пакета дисков], [характер использования цилиндров], [номер участка в файле], адрес участка в пакете, количество дорожек в участке, [дорожка разделения цилиндров].

Параметр "адрес участка в пакете" сообщает операционной системе адрес дорожки пакета, начиная с которой должны размещаться записи файла. Значение параметра вычисляется пользователем как

$$A = 10N_0 + D_0,$$

где N_0 — номер цилиндра, выбранного для размещения начала файла; D_0 — номер начальной дорожки файла в цилиндре с номером N_0 , принимающий значения от 0 до 9.

Параметр "количество дорожек в участке" сообщает операционной системе размеры участка пакета дисков, который отводится под файл. Размер участка представляется количеством содержащихся в файле дорожек. Значение параметра вычисляется пользователем по формуле

$$L = \begin{cases} K/k, & \text{если } K/k - \text{целое число;} \\ E(K/k) + 1, & \text{если } K/k - \text{дробное,} \end{cases}$$

где запись $E(x)$ означает целую часть числа x (см. лабораторную работу 1); K — предельно допустимое количество записей в файле, запланированное программистом при составлении программы; k — количество записей, помещающихся на одной дорожке при обусловленной программой длине каждой записи. Значение k определяется в соответствии с табл. 15.

Таблица 15

Длина записи l , байты	1	2	3	4	6	7	8	9	11	12	14	15	17	19	20
Количество записей на дорожке, k	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45
l	21	23	25	27	30	32	35	37	40	42	45	48	52	55	59
k	44	43	42	41	40	39	38	37	36	35	34	33	32	31	30
l	63	67	72	77	82	88	95	102	109	118	127	138	149		
k	29	28	27	26	25	24	23	22	21	20	19	18	17		
l	162	177	193	213	236	263	295	334	384	447	532				
k	16	15	14	13	12	11	10	9	8	7	6				
l	651	830	1131	1740	3625										
k	5	4	3	2	1										

Примечание. Если заданная длина записи l удовлетворяет условию

$$l_i < l < l_{i+1},$$

где l_i, l_{i+1} — длины записей, указанных соответственно в некоторой i -й и $(i+1)$ -й колонках табл. 15, то количество записей на дорожке определяется как

$$k = k_i,$$

где k_i — количество записей на дорожке, соответствующее длине записи l_i .

Необязательный параметр "дорожка разделения цилиндров" указывает операционной системе номер последней дорожки, занимаемой файлом в каждом из используемых цилиндров в случае размещения файла в пакете дисков с разделением цилиндров. В качестве значения параметра может задаваться любое целое число от 1 до 9. Если цилиндры не разделены, параметр должен быть опущен.

Оператор DEFINE FILE языка программирования ФОРТРАН IV

Назначение оператора. сообщить транслятору с языка ФОРТРАН IV, к какому файлу прямого доступа относится каждое из содержащихся в операторе описаний предельного объема файла и структуры его записей; указать транслятору предельное количество записей в каждом из описываемых файлов и предельную длину каждой записи; сделать доступной для каждого оператора ввода-вывода в исходном модуле, обрабатывающего данный файл, запись, расположенную в файле вслед за записью, обрабатываемой оператором ввода-вывода, логически предшествующим рассматриваемому (т.е. последнему из операторов ввода-вывода, выполненным при работе программы к моменту исполнения рассматриваемого оператора ввода-вывода). Оператор является исполнимым и может получать управление от любого оператора исходного модуля в соответствии с текстом последнего.

Формат оператора следующий:

DEFINE FILE $a_1(m_1, l_1, r_1, t_1)$ [$a_2(m_2, l_2, r_2, t_2)$]...

Параметр a_1 указывает номер файла в ФОРТРАНЕ IV, определяющий логическое устройство ввода-вывода в соответствии с табл. 13. Значением его может быть только целая беззнаковая константа.

Параметр m_1 задает предельное число записей в файле и также принимает значение только в виде целой беззнаковой константы. Параметр l_1 с возможными значениями в виде беззнаковых констант указывает длину каждой записи в байтах или словах (слово соответствует 4 байтам) в зависимости от параметра r_1 . В тех случаях, когда параметру r_1 задано значение L и записи неформатизированные или когда ему задано значение F , значение параметра l_1 указывает длину записи в байтах. Если же значение r_1 равно U и записи неформатизированные, то значение l_1 задает длину записи в словах. Параметр t_1 в любой момент исполнения программы указывает номер записи, непосредственно следующей за последней обработанной записью файла, а в качестве его значения программист может задавать любое символическое имя, описанное тем или иным способом, возможным в ФОРТРАНе IV, как имя целой переменной (последняя доступна для любых операторов программы, логически следующих за данным оператором DEFINE FILE, и называется связанной переменной файла).

П р и м е ч а н и е. Параметры, приведенные в квадратных скобках в описании оператора DEFINE FILE, указывают на возможное, но не обязательное наличие в операторе описаний более чем одного файла.

Подготовка эксплуатируемого пакета дисков при организации нового файла

В соответствии с требуемой структурой файла, предусмотренной оператором DEFINE FILE в проблемной программе, должен быть выполнен шаг задания или отдельное задание на исполнение специальной программы ДОС/ЕС ОЧИСТИТЬ ДИСК, которая производит разметку заданного пользователем участка пакета дисков предварительными записями принятой в создаваемом файле длины. При этом, как уже отмечалось, производится также проверка отсутствия на заданном участке пакета МД файла с истекшим сроком хранения.

Для исполнения этой программы пользователем, как обычно, должны быть подготовлены управляющие операторы DLBL и EXTENT программы УПРАВЛЕНИЕ ЗАДАНИЯМИ. При этом в первом из них в качестве значения параметра "имя файла" должно задаваться UOUT. Кроме того, требуются два специальных управляющих оператора.

Управляющий оператор модификации при разметке файлов с последовательной организацией данных имеет следующий формат:

// UCL B = (K = Ø, D = длина записи), заполнитель, режим

Параметр "длина записи" сообщает операционной системе длину записей, из которых будет состоять создаваемый в проблемной программе файл. Значение параметра должно задаваться в байтах в соответствии с длиной записи, указанной в операторе DEFINE FILE для создаваемого файла.

Параметр "заполнитель" определяет символ, сообщаемый программой ОЧИСТИТЬ ДИСК, который заносится в область ключа и область данных физических записей. Параметр принимает одно из двух значений: C's' и X'xx', где с — произвольный алфавитно-цифровой символ, допустимый в ЭВМ; xx — две шестнадцатиричные цифры.

Параметр "режим" указывает, надо ли производить проверку разметки заданного участка пакета посредством контрольного чтения. При значении параметра OY проверка будет производиться, при значении ON — не будет.

Управляющий оператор END указывает на конец потока управляющих операторов программы ОЧИСТИТЬ ДИСК. Его формат

// END

Оператор END должен быть расположен в задании сразу за оператором UCL, а последний — за оператором EXEC, соответствующим программе ОЧИСТИТЬ ДИСК.

Программа каталогизирована в системной библиотеке абсолютных модулей под именем CLRDSK.

Пример 1. Составить задание на разметку участка диска, предназначенного для размещения выводного файла, занимающего 30 дорожек, начиная с дорожки 1046. Длина каждой записи 16 байтов. В качестве заполнителя принять символ Z; предусмотреть контрольное чтение.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB ОЧСТДИСК
// DLBL UOUT,'СОЗДАВАЕМЫЙ ФАЙЛ',Ø
// EXTENT SYSØØ3,Ø,1Ø46,3Ø
// ASSGN SYSØØ3,X'191'
// EXEC CLRDSK
// UCL B=(K=Ø,D=16),C'Z',OY
// END
/&
```

Пример 2. Составить задание на исполнение программы, каталогизированной в личной библиотеке абсолютных модулей под именем REGIM и использующей выводной файл с порядковым номером 6 на ФОРТРАНе IV и идентификатором 'МАТРИЦА_З_УЗ_ЛОВОЕ', если файл должен быть размещен на том же пакете дисков, что и личная библиотека абсолютных модулей, начиная с 6-й дорожки 104-го цилиндра, должен занимать 30 дорожек и иметь срок сохранности до 31 декабря 1988 г.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB ВЫВДФАЙЛ
// ASSGN SYSC'LB,X'191'
// ASSGN SYSØØ3,X'191'
// DLBL USYSØ3,'МАТРИЦА_З_УЗ_ЛОВОЕ',88/336,SD
// EXTENT SYSØØ3,1,Ø,1Ø46,3Ø
// EXEC REGIM
// (Файл с данными)
/*
/&
ASSGN SYSC'LB,UA
```

2. ПРОГРАММА POSHARST ДЛЯ СОЗДАНИЯ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК ЭЛЕКТРОСТАНЦИЙ ЭНЕРГОСИСТЕМЫ

Описание задачи

Для каждой станции энергосистемы по заданному составу включенных в работу агрегатов и их расходным характеристикам строится расходная характеристика, соответствующая оптимальному распределению между агрегатами любой допустимой суммарной нагрузки станции. Эта характеристика записывается в файл прямого доступа, предназначенный для хранения наборов расходных характеристик, соответствующих наиболее часто встречающимся на станциях энергосистемы составам включенных в работу агрегатов. Характеристики, записанные в файле, идентифицируются для последующей выборки соответствующими им составами агрегатов, которые для ускорения поиска нужной характеристики упорядочиваются программным путем по возрастанию эксплуатационных номеров агрегатов. По желанию пользователя может производиться печать характеристик либо одновременно с их записью в файле, либо вместо последней.

Программа решения задачи

Описание. Программа POSHARST написана на языке ФОРТРАН IV и состоит из основной программы POSHARST и трех подпрограмм HARST, DYNPRG, PECHAT. Подпрограмма HARST имеет две точки входа — HARST и HARST1. Подпрограмма DYNPRG описана в лабораторной работе 1. Структу-

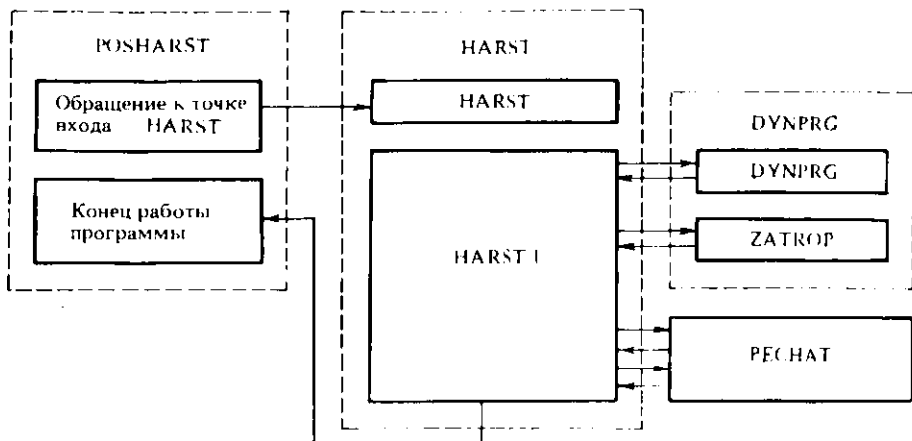


Рис. 19.

ра программы приведена на рис. 19. Тексты исходных модулей программы POSHARST и подпрограмм HARST и PECHAT приводятся в приложении.

Возможности. Программа POSHARST позволяет:

1) организовать файл прямого доступа для хранения расходных характеристик электростанций энергосистемы, которая должна содержать не более 50 станций, причем количество расходных характеристик для каждой станции, соответствующее различным составам включенного на ней в работу оборудования, не превышает 20;

2) записать в файл по одной расходной характеристике, соответствующей заданному составу включенного в работу оборудования, для каждой станции с представлением характеристик в виде ломаных линий, каждая из которых содержит не более 50 точек излома, отстоящих друг от друга на величину шага по мощности n_i

$$\Delta P_i = E \left(\frac{\sum_{j=1}^{n_i} (P_{j \max} - P_{j \min})}{49} \right) + 1,$$

причем единицы измерения расхода топлива, которые должны быть одинаковыми для всех станций, задаются пользователем;

3) идентифицировать области файла, содержащие расходные характеристики различных станций, порядковыми номерами, присваиваемыми станциям программным путем в соответствии с порядком расположения относящихся к ним исходных данных в колоде перфокарт;

4) идентифицировать каждую расходную характеристику соответствующим ей набором эксплуатационных номеров агрегатов станции в порядке их возрастания;

5) обеспечить условия точности оптимального распределения текущих значений суммарной нагрузки каждой станции между ее агрегатами. Эти условия должны соответствовать управляющим параметрам, предусмотренным программой RASNAGAI (для программы POSHARST условия точности должны быть одинаковыми для всех станций);

6) задавать по желанию пользователя режимы работы, соответствующие следующим требованиям, указанным с помощью управляющей карты: только отпечатать характеристики; только запомнить характеристики; отпечатать и запомнить характеристики.

Контрольный пример

Для иллюстрации возможностей программы POSHARST, форм ее входной и выходной информации, а также кодирования исходных данных по задаче рассматривается решение контрольного примера построения расходных характеристик станции. Образцы кодирования данных контрольного примера приведены в таблицах, представляющих формы карт данных и управления программы POSHARST (в графах "Содержимое в контрольном примере"). Полностью исходные данные контрольного примера приведены в параграфе "Ввод".

Ввод

Карты управления. Программа предусматривает использование такого же набора карт управления (кроме карты ККА), как и программа RASNAGA1, который должен быть дополнен картой количества станций ККС (табл. 16) и картой настройки на режим КННР (табл. 17).

Карты управления ККА подготавливаются для каждой станции и должны предшествовать картам наборов данных электростанций (рис. 20).

Карты данных. Набор карт данных для программы POSHARST состоит из наборов карт данных для каждой из электростанций (НКДЭ) системы, подготавливаемых в таком же виде, как и для программы RASNAGA1, за исключением того, что карты КЗМС не должны входить в эти наборы.

Таблица 16

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Количество станций	5
4-79	Пробелы	
80	Порядковый номер перфокарты в колоде	1

Таблица 17

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-38	Требуемый режим	ОТПЕЧАТАТЬ ☐ И ☐ ЗАПОМНИТЬ ☐
39-79	Пробелы	ХАРАКТЕРИСТИКИ ☐
80	Порядковый номер перфокарты в колоде	4

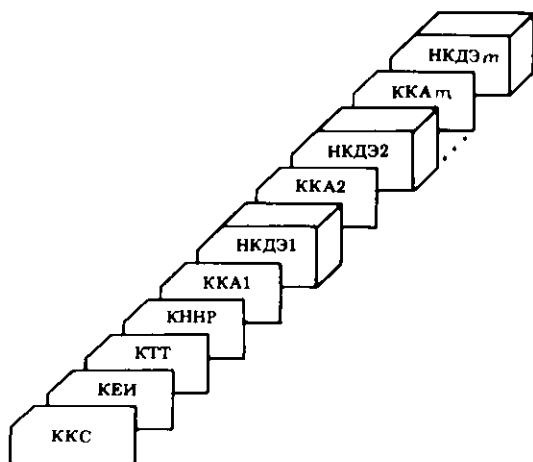


Рис. 20.

Примечание. В контрольном примере карты управления и данных, назначение которых такое же, как и в программе RASNAGA1, имеют форматы, описанные в лабораторной работе 1.

Колода перфокарт. Общий вид колоды перфокарт для работы программы, включая карты управления и карты данных, показан на рис. 20. Распечатка вводимых карт для контрольного примера приведена на рис. 21.

Вывод

Выходная информация программы POSHARST определяется режимом ее работы и имеет следующий вид.

1. Режим **ТОЛЬКО ОТПЕЧАТАТЬ ХАРАКТЕРИСТИКИ**. Для каждой станции выводятся на печать присвоенный ей номер, состав включенных в работу агрегатов и расходная характеристика в удобном для восприятия виде, показанном на рис. 22.

2. Режим **ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ**. Вывод данных на печать проблемной программой отсутствует. Пользователь при обработке операционной системой задания на выполнение программы получает на устройстве печати только стандартное сообщение системы об окончании обработки этого задания.

3. Режим **ОТПЕЧАТАТЬ И ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ**. Вывод на печать такой же, как в пункте 1.

```

// JOB РАСПЧТКА
// ASSEM SYS004, X, 0000
// ASSEM SYS005, X, 0000
// EXEC CDP
// UCP TD, FF, A=(000, 000), B=(1200), Q=(77, 4)
// M1
// END

```

РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ

CARD TO PRINTER / PUNCH UTILITY

INPUT RECORD LENGTH 0000

OUTPUT RECORD LENGTH 0120

INPUT BLOCK LENGTH 0000

OUTPUT BLOCK LENGTH 00120

INPUT OPTION CARD 000

OUTPUT OPTION PRINT CHARACTER

2 INPUT, 2 OUTPUT AREAS ASSIGNED

RECORD FORMAT FIXED

TYPE DATA DISPLAY

STARTING SEQUENCE COLUMN 77

SEQUENCE LENGTH 04

STARTING RECORD NUMBER 00000001

РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ

BL	SZ	BL	NO	RC	NO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	5	1	5	1	5	1	5	1	5	1	5	1	5	1	5	1	5	1	5	1	5	1
2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2
3	5	3	5	3	5	3	5	3	5	3	5	3	5	3	5	3	5	3	5	3	5	3
4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	5	6	5	6	5	6	5	6	5	6	5	6	5	6	5	6	5	6	5	6	5	6
7	5	7	5	7	5	7	5	7	5	7	5	7	5	7	5	7	5	7	5	7	5	7
8	5	8	5	8	5	8	5	8	5	8	5	8	5	8	5	8	5	8	5	8	5	8
9	5	9	5	9	5	9	5	9	5	9	5	9	5	9	5	9	5	9	5	9	5	9
10	5	10	5	10	5	10	5	10	5	10	5	10	5	10	5	10	5	10	5	10	5	10
11	5	11	5	11	5	11	5	11	5	11	5	11	5	11	5	11	5	11	5	11	5	11
12	5	12	5	12	5	12	5	12	5	12	5	12	5	12	5	12	5	12	5	12	5	12
13	5	13	5	13	5	13	5	13	5	13	5	13	5	13	5	13	5	13	5	13	5	13
14	5	14	5	14	5	14	5	14	5	14	5	14	5	14	5	14	5	14	5	14	5	14
15	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15
16	5	16	5	16	5	16	5	16	5	16	5	16	5	16	5	16	5	16	5	16	5	16
17	5	17	5	17	5	17	5	17	5	17	5	17	5	17	5	17	5	17	5	17	5	17

Рис. 21 (начало).

18	18	80.Е0	17.Е0	90.Е0	21.5Е0	35.Е0	6.25Е0	19
19	19	20.Е0	4.Е0	30.Е0	5.5Е0			19
20	20	45.Е0	10.25Е0	50.Е0	12.25Е0			20
21	21	35.Е0	7.Е0	45.Е0	9.Е0	55.Е0	11.Е0	21
22	22	70.Е0	17.75Е0	80.Е0	22.25Е0			22
23	23	50.Е0	10.Е0	60.Е0	11.7Е0	70.Е0	13.4Е0	23
24	24	80.Е0	16.9Е0	100.Е0	23.9Е0			24
25	25	40.Е0	8.Е0	50.Е0	9.8Е0	60.Е0	11.6Е0	25
26	26	70.Е0	15.6Е0	90.Е0	23.6Е0			26
27	27	50.Е0	10.Е0	60.Е0	11.5Е0	70.Е0	13.Е0	27
28	28	80.Е0	16.Е0	90.Е0	19.Е0	1.Е2	23.5Е0	28
29	29	30.Е0	6.Е0	40.Е0	8.Е0	45.Е0	9.Е0	29
30	30	50.Е0	11.Е0	60.Е0	15.Е0	70.Е0	21.Е0	30
31	31	75.Е0	24.Е0					31
32	32	10						32
33	33	1	2	4	6	8	4	33
34	34	10.Е0	2.Е0	12.Е0	2.4Е0	14.Е0	2.8Е0	34
35	35	17.Е0	4.3Е0	20.Е0	5.8Е0			35
36	36	40.Е0	8.Е0	50.Е0	9.5Е0	60.Е0	11.Е0	36
37	37	70.Е0	16.5Е0	80.Е0	22.Е0			37
38	38	30.Е0	6.Е0	40.Е0	8.5Е0	50.Е0	11.Е0	38
39	39	65.Е0	20.Е0	75.Е0	26.Е0			39
40	40	25.Е0	5.Е0	30.Е0	5.5Е0	40.Е0	6.5Е0	40
41	41	50.Е0	10.5Е0	60.Е0	14.5Е0			41
42	42	30.Е0	6.Е0	40.Е0	7.3Е0	50.Е0	8.6Е0	42
43	43	60.Е0	13.0Е0	70.Е0	19.2Е0	80.Е0	24.5Е0	43
44	44	40.Е0	8.Е0	50.Е0	9.5Е0	70.Е0	12.5Е0	44
45	45	80.Е0	17.Е0	90.Е0	21.5Е0			45
46	46	20.Е0	4.Е0	30.Е0	5.5Е0	35.Е0	6.25Е0	46
47	47	45.Е0	10.25Е0	50.Е0	12.25Е0			47
48	48	35.Е0	7.Е0	45.Е0	9.Е0	55.Е0	11.Е0	48
49	49	70.Е0	17.75Е0	80.Е0	22.25Е0			49
50	50	50.Е0	10.Е0	60.Е0	11.7Е0	70.Е0	13.4Е0	50
51	51	80.Е0	16.9Е0	100.Е0	23.9Е0			51
52	52	40.Е0	8.Е0	50.Е0	9.8Е0	60.Е0	11.6Е0	52
53	53	70.Е0	15.6Е0	90.Е0	23.6Е0			53
54	54	7						54
55	55	1	2	4	6	8	4	55

Рис. 21 (продолжение).

[illegible]

Рис. 21 (продолжение).

92	92	10.ЕВ	2.ЕВВ	12.ЕВ	2.4ЕВ	14.ЕВ	2.РЕВ	92
93	93	17.ЕВ	4.3ЕВ	20.ЕВ	5.8ЕВ	60.ЕВ	11.ЕВ	93
94	94	40.ЕВ	8.ЕВВ	50.ЕВ	9.5ЕВ	60.ЕВ	11.ЕВ	94
95	95	70.ЕВ	16.5ЕВ	80.ЕВ	22.ЕВ	50.ЕВ	11.ЕВ	95
96	96	30.ЕВ	6.ЕВВ	40.ЕВ	8.5ЕВ	50.ЕВ	11.ЕВ	96
97	97	65.ЕВ	20.ЕВ	75.ЕВ	26.ЕВ	40.ЕВ	6.5ЕВ	97
98	98	25.ЕВ	5.ЕВВ	30.ЕВ	5.5ЕВ	40.ЕВ	6.5ЕВ	98
99	99	50.ЕВ	10.5ЕВ	60.ЕВ	14.5ЕВ	30.ЕВ	8.6ЕВ	99
100	100	30.ЕВ	6.ЕВВ	40.ЕВ	7.3ЕВ	50.ЕВ	24.5ЕВ	100
101	101	60.ЕВ	13.9ЕВ	70.ЕВ	19.2ЕВ	80.ЕВ	12.5ЕВ	101
102	102	40.ЕВ	8.ЕВВ	50.ЕВ	9.5ЕВ	70.ЕВ	6.25ЕВ	102
103	103	80.ЕВ	17.ЕВ	90.ЕВ	21.5ЕВ	35.ЕВ	11.ЕВ	103
104	104	20.ЕВ	4.ЕВВ	30.ЕВ	5.5ЕВ	35.ЕВ	13.4ЕВ	104
105	105	45.ЕВ	10.25ЕВ	50.ЕВ	12.25ЕВ	55.ЕВ	11.6ЕВ	105
106	106	35.ЕВ	7.ЕВВ	45.ЕВ	9.ЕВВ	55.ЕВ	13.ЕВ	106
107	107	70.ЕВ	17.75ЕВ	80.ЕВ	22.25ЕВ	70.ЕВ	21.ЕВ	107
108	108	50.ЕВ	10.ЕВ	60.ЕВ	11.7ЕВ	60.ЕВ	13.ЕВ	108
109	109	80.ЕВ	16.9ЕВ	100.ЕВ	23.9ЕВ	60.ЕВ	11.6ЕВ	109
110	110	40.ЕВ	8.ЕВВ	50.ЕВ	9.8ЕВ	60.ЕВ	13.ЕВ	110
111	111	70.ЕВ	15.6ЕВ	90.ЕВ	23.6ЕВ	70.ЕВ	23.5ЕВ	111
112	112	50.ЕВ	10.ЕВ	60.ЕВ	11.5ЕВ	70.ЕВ	9.ЕВВ	112
113	113	80.ЕВ	16.ЕВ	90.ЕВ	19.ЕВ	1.ЕВ	45.ЕВ	113
114	114	30.ЕВ	6.ЕВВ	40.ЕВ	8.ЕВВ	45.ЕВ	21.ЕВ	114
115	115	50.ЕВ	11.ЕВ	60.ЕВ	15.ЕВ	70.ЕВ	116	115
116	116	75.ЕВ	24.ЕВ					116

END OF DATA
PAGE 28L NO RC NO.....5.....1.....5.....2.....5.....3.....5.....4.....5.....5.....6.....5.....7.....5.....8.....5.....
NUMBER OF INPUT BLOCKS PROCESSED 000116
NUMBER OF OUTPUT BLOCKS PROCESSED 000116
END OF JOB
EOJ РАСЧЕТКА РАСПЕЧАТКА ВЫПОЛНЕНА
11.43 01.DURATION 00.02.26

Рис. 21 (окончание).

Нагрузка станции, МВт	1	2	3	4	5	6	7	8	9	10
	0310.741E	0310.752E	0310.763E	0310.774E	0310.785E	0310.796E	0310.807E	0310.818E	0310.829E	0310.840E
Расход топлива т у.т.	1	1	1	1	1	1	1	1	1	1
	0310.158E	0310.162E	0310.167E	0310.172E	0310.176E	0310.181E	0310.186E	0310.191E	0310.197E	0310.203E
Нагрузка станции, МВт	1	1	1	1	1	1	1	1	1	1
	0310.840E	0310.851E	0310.862E	0310.873E	0310.884E	0310.895E	0310.906E	0310.917E	0310.928E	0310.939E
Расход топлива т у.т.	1	1	1	1	1	1	1	1	1	1
	0310.209E	0310.215E	0310.221E	0310.228E	0310.234E	0310.241E	0310.246E	0310.252E	0310.258E	0310.264E

РАСХОДАЯ ХАРАКТЕРИСТИКА СТАНЦИИ N 2

ВКЛЮЧЕНЫ В РАБОТУ АГРЕГАТЫ №№: 1 2 3 4 5 6 7 8 9 10

Нагрузка станции, МВт	1	2	3	4	5	6	7	8	9	10
	0310.320E	0310.329E	0310.338E	0310.347E	0310.356E	0310.365E	0310.374E	0310.383E	0310.392E	0310.401E
Расход топлива т у.т.	1	1	1	1	1	1	1	1	1	1
	0310.640E	0310.649E	0310.658E	0310.667E	0310.676E	0310.685E	0310.694E	0310.703E	0310.712E	0310.721E

Рис. 22 (продолжение).

РАСХОДНАЯ ХАРАКТЕРИСТИКА СТАНЦИИ № 3

ВКЛЮЧЕНЫ В РАБОТУ АГРЕГАТЫ №: 1 2 3 4 5 6 7

Нагрузка станции МВт	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е
10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е
10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е
10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е	10.195Е
10.195Е	10.195Е	10.195																										

Инструкция по применению программы

Программа POSHARST каталогизирована в личную библиотеку абсолютных модулей. Задание на исполнение программы может иметь, например, следующий вид (в режиме ТОЛЬКО ОТПЕЧАТАТЬ ХАРАКТЕРИСТИКИ управляющие операторы, необходимые для работы с файлом, могут отсутствовать):

```
// JOB ОТКРФАЙЛ 'ОТКРЫТИЕ_ФАЙЛА_РАСХОДНЫХ_
ХАРАКТЕРИСТИК'
ASSGN SYSC'LB,X'191'
// ASSGN SYSØØ3,X'191'
// DLBL PSYSØØ3,'ФАЙЛ_РАСХОДНЫХ_ХАРАКТЕРИСТИК
_СТАНЦИЙ'
// EXTENT SYSØØ3,111111,,,11,1ØØ
// EXEC POSHARST
(Файл карт управления и данных)
/*
ASSGN SYSC'LB, UA
/&_ _ _ _ _ РЕШЕНИЕ_ЗАДАЧИ_ДЮ_ВАРИАНТУ_ЛН
_ ВЫПОЛНЕНО
```

Примечание. В позицию 42 карты /& заносится номер указанного преподавателем варианта задания на выполнение лабораторной работы (см. раздел 7 данной лабораторной работы).

Сообщения об ошибках и время решения

Во время работы программы могут выдаваться только стандартные сообщения операционной системы об имеющихся ошибках.

Время решения контрольного примера программой POSHARST для каждого из ее возможных режимов составляет соответственно 2 мин 49 с, 2 мин 06 с, 2 мин 51 с.

Подпрограмма HARST

Назначение. Выполняет все функции программы POSHARST, но оформлена в виде подпрограммы, чтобы обеспечить возможность автоматического подключения к другим объектным модулям, транслированным с ФОРТРАНа IV (в частности, к программе коррекции файла расходных характеристик станций CORHAR, рассматриваемой в лабораторной работе 4). Для этой цели, кроме того, созданы две точки входа в программу:

- а) HARST, к которой производится обращение в программе POSHARST;
- б) HARST1, к которой обращается программа CORHAR.

Точка входа HARST обеспечивает построение расходных характеристик для всех электростанций системы, а точка входа HARST1 — только для тех станций, на которых планируется включение в работу таких составов оборудования, что соответствующие им расходные характеристики отсутствуют в файле.

Обращение. а) CALL HARST (NOMHAR);

б) CALL HARST1 (KOLST, NOMST, NOMHAR).

Описание параметров. NOMHAR — порядковый номер записываемой в файл расходной характеристики станции, указывающий ее расположение в

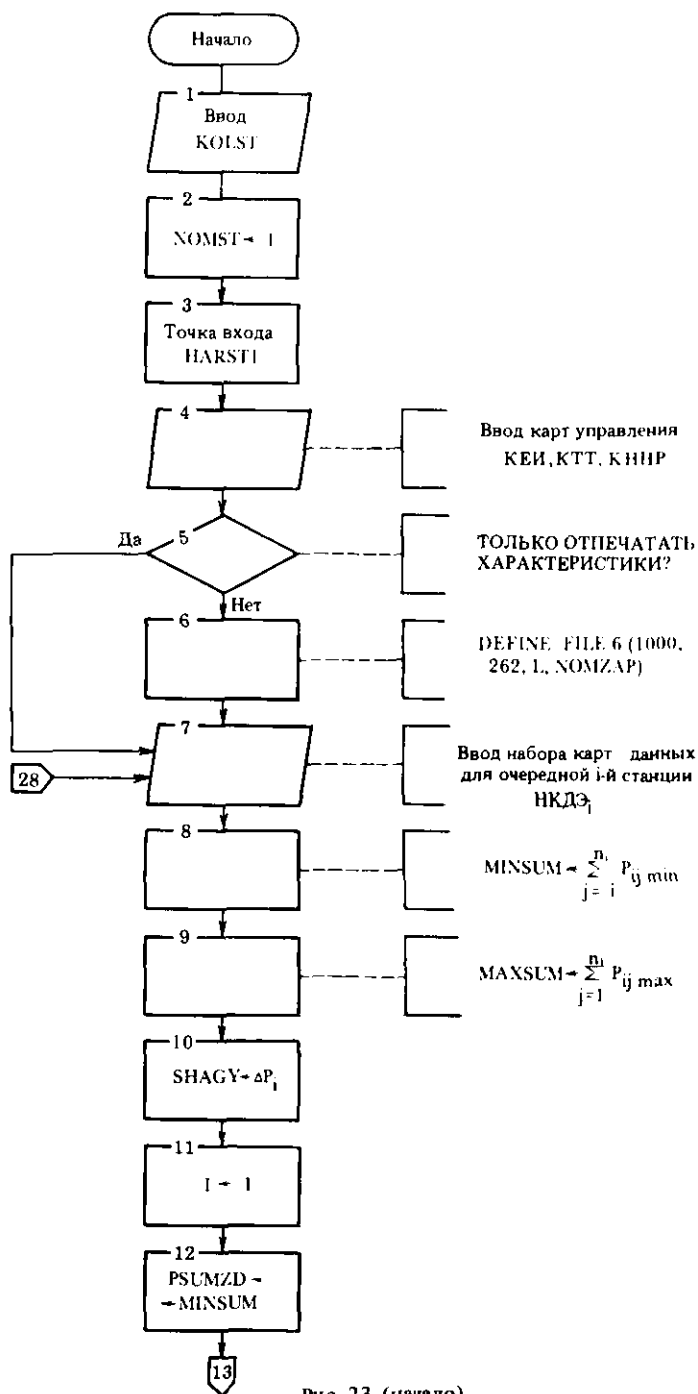


Рис. 23 (начало).

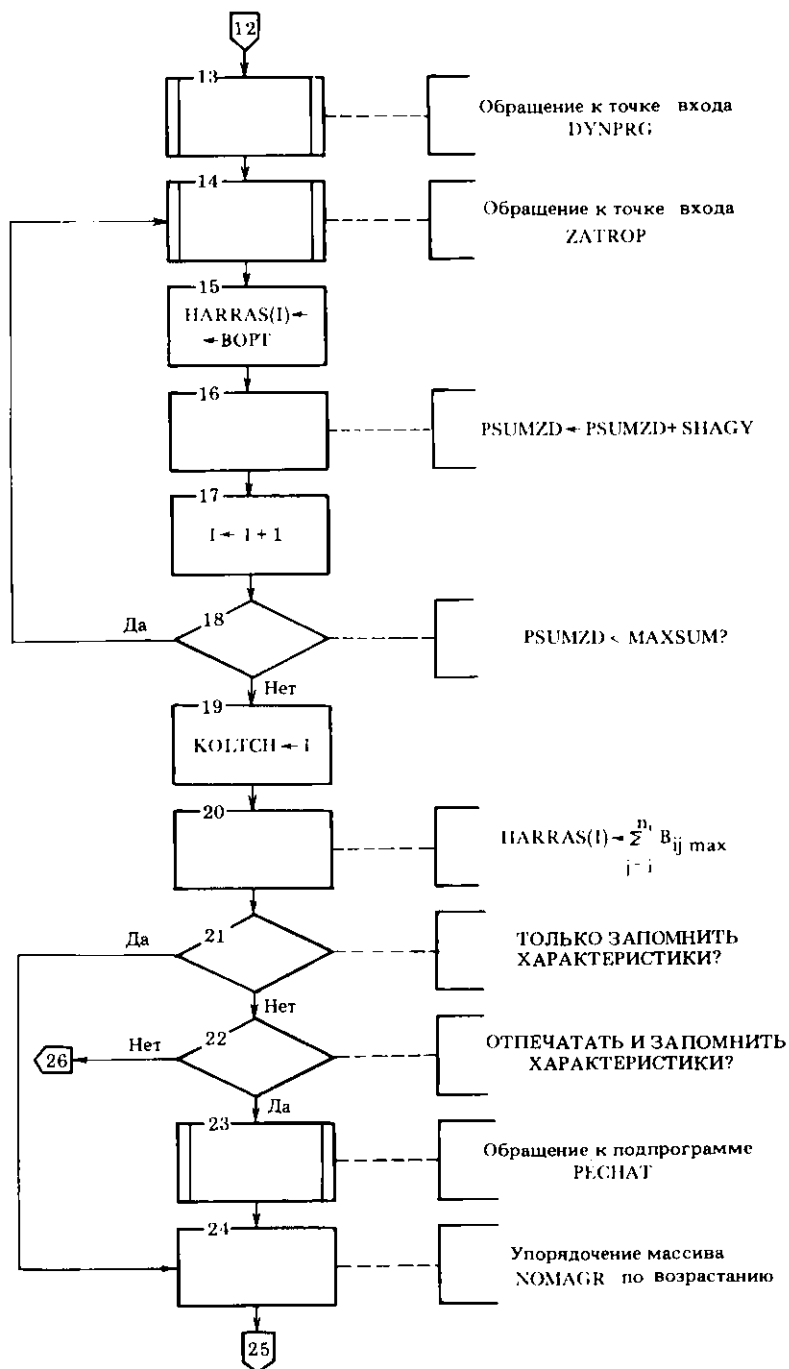


Рис. 23 (продолжение).

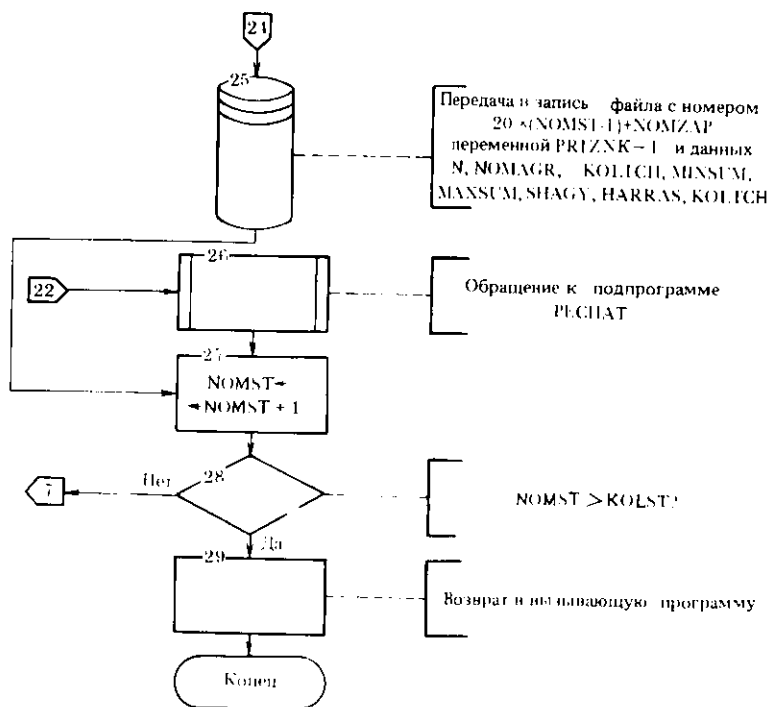


Рис. 23 (окончание).

файле по отношению к другим характеристикам данной станции; KOLST — количество станций, для которых требуется пополнение файла новой характеристикой; NOMST — порядковый номер области файла, отведенной под расходные характеристики станции.

Метод. Метод построения расходных характеристик станций описан в разделе "Краткие теоретические сведения". Схема подпрограммы приведена на рис. 23.

Примечания. 1. На рис. 23 используются обозначения, принятые в лабораторной работе 1.

2. Имена фактических аргументов обращений подпрограммы HARST к подпрограмме DYNPRG, используемые в схеме, приняты совпадающими с именами соответствующих фактических аргументов, приведенных при описании DYNPRG в лабораторной работе 1.

3. Переменная PRIZNK при значении, равном 0, указывает, что запись является последней в области файла для i -й станции, а при значении 1 — что запись последней не является.

Подпрограмма DYNPRG

Вся информация, достаточная для того, чтобы можно было использовать подпрограмму при разработке новых программ, приведена в лабораторной работе 1.

Подпрограмма РЕСНАТ

Назначение. Осуществляет печать результатов решения задачи в соответствующих режимах работы программы POSHARST. Ввиду узкого целевого назначения подпрограммы, специально предназначенной для работы в составе программ POSHARST и CORHAR, подробное описание подпрограммы не приводится. По своей логической структуре подпрограмма близка к участку подпрограммы DYNPRG с точкой входа VIVOD.

3. ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

В данной лабораторной работе ставится задача освоения методики подготовки пакета магнитных дисков к выводу файла прямого доступа и составления задания на исполнение проблемных программ, написанных на ФОРТРА-Не IV и включающих обработку файлов прямого доступа, на примере POSHARST. Соответственно при подготовке к выполнению лабораторной работы необходимо:

1) убедиться, что в достаточной степени усвоено содержание ранее выполненных лабораторных работ 1, 2;

2) изучить раздел 1 данной лабораторной работы, обратив особое внимание на методику построения расходных характеристик электростанций методом динамического программирования и методику задания значений параметров управляющих операторов EXTENT, DLBL и их взаимодействие с оператором DEFINE FILE ФОРТРАНа IV, на правила назначения логических и физических устройств ввода-вывода для хранения файлов на магнитных дисках, на составление заданий специальной системной программе ОЧИСТИТЬ ДИСК и проблемной программе;

3) изучить раздел 2 данной лабораторной работы, обратив особое внимание на правила подготовки исходных данных для работы программы POSHARST и структуру создаваемого файла в соответствии с оператором DEFINE FILE, показанным на схеме подпрограммы HARST (см. рис. 23);

4) получить у преподавателя один из вариантов задания на выполнение данной лабораторной работы;

5) подготовить исходную информацию по предложенному варианту задания на бланках для сдачи на перфорацию.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать следующие разделы:

- 1) цель работы;
- 2) методику декомпозиции задачи распределения активных мощностей в энергосистеме при использовании метода динамического программирования;
- 3) схемы программы POSHARST и подпрограммы HARST;
- 4) задание операционной системе для соответствующего варианта исходных данных;

- 5) сообщения операционной системы для соответствующего варианта о результатах обработки задания;
- 6) графики расходных характеристик электростанций.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что означает понятие "файл данных"?
2. Почему целесообразно производить декомпозицию задачи оптимального распределения активных мощностей в энергосистеме на подзадачи внутри- и межстанционной оптимизации?
3. Как формируется семейство задач оптимизации для построения расходной характеристики электростанции при заданном составе включенных в работу агрегатов?
4. Чем обеспечивается эффективность метода динамического программирования при построении расходной характеристики электростанции?
5. С какой целью создается файл расходных характеристик электростанций энергосистемы?
6. В чем состоит отличие понятий физической и логической записей файла?
7. Чем отличаются методы последовательного и прямого доступа к записям файла?
8. Почему файлы прямого доступа создаются только на магнитных дисках?
9. В каких случаях используются файлы с неформатизованными либо с форматизованными записями?
10. Какие ограничения должны учитываться при выборе номеров создаваемых файлов в процессе составления программы на языке ФОРТРАН IV?
11. Как производится назначение физических устройств ввода-вывода логическим устройствам при обработке в одном задании для ДОС/ЕС нескольких файлов, размещенных на одном пакете магнитных дисков?
12. Для чего создаются метки файлов и какими средствами располагает пользователь для их создания?
13. Как осуществляется защита файла от преждевременного разрушения?
14. Чем различается использование управляющих операторов DLBL и EXTENT при обработке вводных и выводных файлов?
15. Каким образом пользователь задает в управляющих операторах информацию об участке пакета дисков, отводимом под файл в соответствии с требованиями проблемной программы?
16. Почему для подготовки пакета дисков к работе необходимо предусмотреть отдельное задание либо шаг задания на исполнение специальной системной программы ОЧИСТИТЬ ДИСК?
17. Каким образом программист может обеспечить пользователю возможность выбора желаемого режима работы проблемной программы, составленной на языке ФОРТРАН IV?
18. Какие функции осуществляет программа POSHARST?
19. Почему важная часть программы POSHARST оформлена в виде подпрограммы HARST?

20. Почему в подпрограмме HARST использовано обращение к двум различным точкам входа подпрограммы DYNPRG?

21. Для чего при идентификации расходных характеристик станций в файле предусмотрено упорядочение эксплуатационных номеров агрегатов по их возрастанию?

6. ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Вариант 1

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-7	$\frac{1}{5}$	$\frac{3}{5}$	$\frac{5}{4}$	$\frac{7}{4}$		
ТЭЦ-4	$\frac{6}{3}$	$\frac{4}{3}$	$\frac{3}{2}$	$\frac{8}{1}$	$\frac{2}{2}$	
ТЭЦ-3	$\frac{2}{1}$	$\frac{4}{1}$	$\frac{7}{3}$	$\frac{3}{1}$	$\frac{5}{3}$	$\frac{6}{3}$
ГРЭС-5	$\frac{5}{5}$	$\frac{4}{5}$	$\frac{7}{6}$			

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле – 1912-я дорожка; $\Delta Y^* = 0,5\%$; $\Delta Y_{\text{max доп}} = 5 \text{ МВт}$.

Вариант 2

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ТЭЦ-2	$\frac{2}{2}$	$\frac{1}{2}$	$\frac{5}{1}$	$\frac{4}{2}$	$\frac{7}{1}$	
ГРЭС-1	$\frac{6}{4}$	$\frac{3}{5}$	$\frac{5}{4}$	$\frac{1}{5}$		
ТЭЦ-1	$\frac{8}{1}$	$\frac{3}{3}$	$\frac{1}{3}$	$\frac{5}{1}$	$\frac{6}{1}$	$\frac{4}{1}$
ГРЭС-2	$\frac{2}{4}$	$\frac{7}{6}$	$\frac{4}{4}$	$\frac{6}{4}$		

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле – 1922-я дорожка; $\Delta Y^* = 0,8\%$; $\Delta Y_{\text{max доп}} = 9 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-5	$\frac{4}{6}$	$\frac{3}{5}$	$\frac{6}{6}$			
ТЭЦ-2	$\frac{6}{1}$	$\frac{2}{2}$	$\frac{5}{1}$	$\frac{7}{1}$	$\frac{1}{2}$	
ГРЭС-2	$\frac{5}{4}$	$\frac{3}{5}$	$\frac{7}{4}$	$\frac{4}{5}$		
ТЭЦ-4	$\frac{7}{3}$	$\frac{2}{2}$	$\frac{6}{3}$	$\frac{1}{2}$	$\frac{8}{3}$	$\frac{9}{1}$
ГРЭС-3	$\frac{1}{4}$	$\frac{5}{5}$	$\frac{7}{5}$			

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле – 1932-я дорожка; $\Delta Y^* = 0,5 \%$; $\Delta Y_{\text{max доп}} = 10 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-5	$\frac{3}{2}$	$\frac{4}{4}$	$\frac{1}{2}$	$\frac{5}{4}$		
ТЭЦ-8	$\frac{7}{1}$	$\frac{2}{2}$	$\frac{5}{2}$	$\frac{3}{2}$	$\frac{9}{1}$	
ГРЭС-3	$\frac{3}{4}$	$\frac{11}{6}$	$\frac{9}{6}$			
ТЭЦ-5	$\frac{2}{3}$	$\frac{5}{3}$	$\frac{11}{1}$	$\frac{7}{1}$	$\frac{9}{1}$	$\frac{10}{1}$

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле – 1942-я дорожка; $\Delta Y^* = 0,7 \%$; $\Delta Y_{\text{max доп}} = 6 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-8	$\frac{5}{1}$	$\frac{9}{4}$	$\frac{7}{4}$	$\frac{4}{1}$	$\frac{3}{1}$	
ТЭЦ-5	$\frac{6}{3}$	$\frac{4}{3}$	$\frac{7}{1}$	$\frac{9}{1}$	$\frac{8}{1}$	$\frac{1}{2}$
ТЭЦ-4	$\frac{1}{2}$	$\frac{9}{3}$	$\frac{8}{3}$	$\frac{2}{2}$	$\frac{10}{3}$	$\frac{11}{3}$

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле – 1952-я дорожка; $\Delta Y^* = 0,6 \%$; $\Delta Y_{\text{max доп}} = 8 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)			
ТЭЦ-1	$\frac{1}{3}$	$\frac{5}{1}$	$\frac{2}{3}$	$\frac{3}{1}$
ТЭЦ-7	$\frac{7}{2}$	$\frac{2}{2}$	$\frac{8}{1}$	$\frac{4}{2}$ $\frac{10}{1}$
ГРЭС-2	$\frac{5}{6}$	$\frac{3}{5}$	$\frac{1}{5}$	
ГРЭС-9	$\frac{2}{4}$	$\frac{8}{6}$	$\frac{4}{4}$	$\frac{6}{6}$
ГРЭС-5	$\frac{7}{5}$	$\frac{1}{2}$	$\frac{9}{5}$	

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле - 1962-я дорожка; $\Delta Y^* = 0,5\%$; $\Delta Y_{\text{max доп}} = 5 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)						
ТЭЦ-1	$\frac{2}{2}$	$\frac{4}{2}$	$\frac{10}{3}$	$\frac{11}{3}$	$\frac{7}{3}$	$\frac{3}{2}$	$\frac{5}{3}$
ГРЭС-2	$\frac{5}{1}$	$\frac{2}{1}$	$\frac{7}{1}$	$\frac{9}{1}$	$\frac{10}{5}$		
ГРЭС-3	$\frac{11}{1}$	$\frac{8}{1}$	$\frac{3}{2}$	$\frac{4}{2}$	$\frac{2}{3}$		

У к а з а н и е. Задание выполнять при условиях: адрес начала данных в файле - 1972-я дорожка; $\Delta Y^* = 0,7\%$; $\Delta Y_{\text{max доп}} = 2,5 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ТЭЦ-2	$\frac{2}{3}$	$\frac{1}{3}$	$\frac{7}{1}$	$\frac{6}{1}$	$\frac{12}{5}$	$\frac{8}{1}$
ГРЭС-7	$\frac{1}{5}$	$\frac{5}{4}$	$\frac{3}{4}$	$\frac{4}{4}$		
ГРЭС-4	$\frac{2}{4}$	$\frac{7}{6}$	$\frac{3}{4}$			
ТЭЦ-8	$\frac{1}{2}$	$\frac{7}{1}$	$\frac{4}{2}$	$\frac{9}{1}$	$\frac{10}{1}$	

У к а з а н и е. Задание выполнить при условиях: адрес начала данных в файле - 1982-я дорожка; $\Delta Y^* = 0,6\%$; $\Delta Y_{\text{max доп}} = 4,5 \text{ МВт}$.

П р и м е ч а н и е. 1. Типы расходных характеристик агрегатов, подразумеваемые в вариантах заданий, приводятся в табл. 18 (расход топлива выражается в тоннах условного топлива).

2. По каждому варианту необходимо составить задание для операционной системы, включающее исполнение программ ОЧИСТИТЬ ДИСК и POSHARST.

3. В качестве идентификатора файла рекомендуется принять 'ВАРИАНТ Н', где вместо пробела подставляется номер заданного варианта.

4. Число записей в файле с целью экономии места в пакете дисков следует принять отличным от указанного в операторе DEFINE FILE и равным $K=20m$, где m количество станций.

5. Длина каждой записи указана в операторе DEFINE FILE, показанном на схеме подпрограммы HARST на рис. 23.

6. В качестве значения параметра "срок сохранности" следует указать дату окончания семестра, представленную в нужной форме.

7. В качестве параметра "регистрационный номер пакета дисков" необходимо указать набор символов 111111.

8. Текст шага задания на выполнение программы POSHARST необходимо сохранить, так как он понадобится при выполнении лабораторной работы 4.

Таблица 18

Тип характеристики	Координаты точек излома расходной характеристики							
	1		2		3		4	
	Р	В	Р	В	Р	В	Р	В
1	75	27,9	86	29,2	97	31,5	110	33,7
2	50	21,4	57	22,5	65	24,2	73	26,0
3	50	18,5	55	19,7	63	21,0	70	22,7
4	150	48,7	171	51,1	195	55,1	219	59,0
5	130	38,0	160	44,0	190	59,0	200	70,0
6	225	82,0	260	91,0	335	106	425	140,0

Тип характеристики	Координаты точек излома расходной характеристики							
	5		6		7		8	
	Р	В	Р	В	Р	В	Р	В
1	120	36,8	132	40,9	143	49,6	150	58,6
2	80	28,3	88	31,5	95	38,2	100	45,0
3	83	24,8	90	27,6	94	33,2	100	39,5
4	240	64,4	264	71,6	285	86,8	300	102,5
5								
6	500	172,0						

КОРРЕКЦИЯ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК ЭЛЕКТРОСТАНЦИЙ ПРИ ОПЕРАТИВНОЙ ОПТИМИЗАЦИИ РЕЖИМА ЭНЕРГОСИСТЕМЫ

Ц е л ь р а б о т ы: ознакомиться с простейшими приемами пополнения и обновления базы условно-постоянной информации для решения задачи оптимального распределения активных мощностей между электростанциями.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Одной из разновидностей условно-постоянной информации (УПИ) об управляемой производственной системе является совокупность, ретроспективных данных, отражающих характерные ситуации, возникавшие на протяжении ближайшего предшествующего периода функционирования системы. Наличие такой информации, если она представлена в форме, предусматриваемой математическими моделями, на основе которых осуществляется автоматизированное принятие решений по управлению системой, ускоряет получение последних. Это возможно потому, что часть информации, требуемой в процессе решения задачи, во многих случаях оказывается заблаговременно представленной в нужном виде.

Однако охватить в УПИ все случаи, как правило, невозможно из-за того, что число их либо заранее не ограничено, либо настолько велико, что нельзя одновременно хранить в УПИ всю требуемую информацию.

Кроме того, любая производственная система развивается, ввиду чего информация, относящаяся к описываемой предыстории функционирования системы, постепенно устаревает. Поэтому в составе математического обеспечения управляемой базы данных АСУ необходимо предусматривать программы, осуществляющие при необходимости пополнение базы УПИ информацией и ее обновление при исчерпании вместимости соответствующих областей внешней памяти ЭВМ путем записи новой информации на место наиболее устаревшей.

В данной лабораторной работе рассматривается программа CORHAR, осуществляющая указанные функции обслуживания файла расходных характеристик электростанций энергосистемы для различных составов включенных в работу агрегатов. Этот файл используется для ускорения решения одной из важнейших задач оперативного управления энергосистемой — задачи оптимального распределения активных мощностей в системе. Необходимость в такой программе обусловлена тем, что несмотря на конечность числа возможных составов агрегатов в энергосистеме и соответственно числа возможных расходных характеристик станций, упомянутый файл практически не может содержать все возможные расходные характеристики.

Действительно, если на m электростанций энергосистемы установлено $n_1, n_2, \dots, n_m$ агрегатов, то число возможных составов включенного в работу оборудования равно:

$$S = \prod_{i=1}^m (2^{n_i} - 1),$$

а число возможных характеристик станций энергосистемы

$$h = \sum_{i=1}^m (2^{n_i} - 1)$$

и весьма велико для энергосистем больших размеров.

Поэтому практически можно организовать только файл, содержащий часть всех возможных характеристик станций, соответствующую составам включенных в работу агрегатов, встречавшихся на протяжении некоторого предшествующего периода эксплуатации.

Организация файла начинается с записи в соответствующие области пакета магнитных дисков по одной характеристике для каждой станции, что осуществляется программой POSHARST, рассмотренной в лабораторной работе 3, а для текущей коррекции файла необходима программа CORHAR.

2. ПРОГРАММА CORHAR ДЛЯ КОРРЕКЦИИ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК ЭЛЕКТРОСТАНЦИЙ ЭНЕРГОСИСТЕМЫ

Описание задачи

Программа позволяет по желанию пользователя выполнить разнообразные функции обслуживания файла для заданного подмножества электростанций энергосистемы: проверить, имеется ли в файле расходная характеристика станции для указанного пользователем состава включенных в работу агрегатов; сообщить пользователю, в какую область файла следует записывать вновь формируемую характеристику станции; отпечатать нужную характеристику; сообщить, каким составам агрегатов соответствуют характеристики станций, имеющиеся в файле; построить и записать в файл характеристику станции для заданного состава агрегатов.

Программа решения задачи

Описание. Программа CORHAR написана на языке ФОРТРАН IV и состоит из основной программы CORHAR и трех подпрограмм HARST, DYNPRG, PECHAT, описанных в лабораторной работе 1. Схема программы приведена на рис. 24. Текст исходного модуля программы CORHAR приведен в приложении.

Возможности. При ограничениях по точности решения и предельному объему решаемой задачи, аналогичных указанным в программе POSHARST, программа CORHAR для заданного подмножества станций энергосистемы позволяет:

1) осуществить поиск в файле прямого доступа на пакете магнитных дисков расходных характеристик станций для заданных на них составов включенных в работу агрегатов.

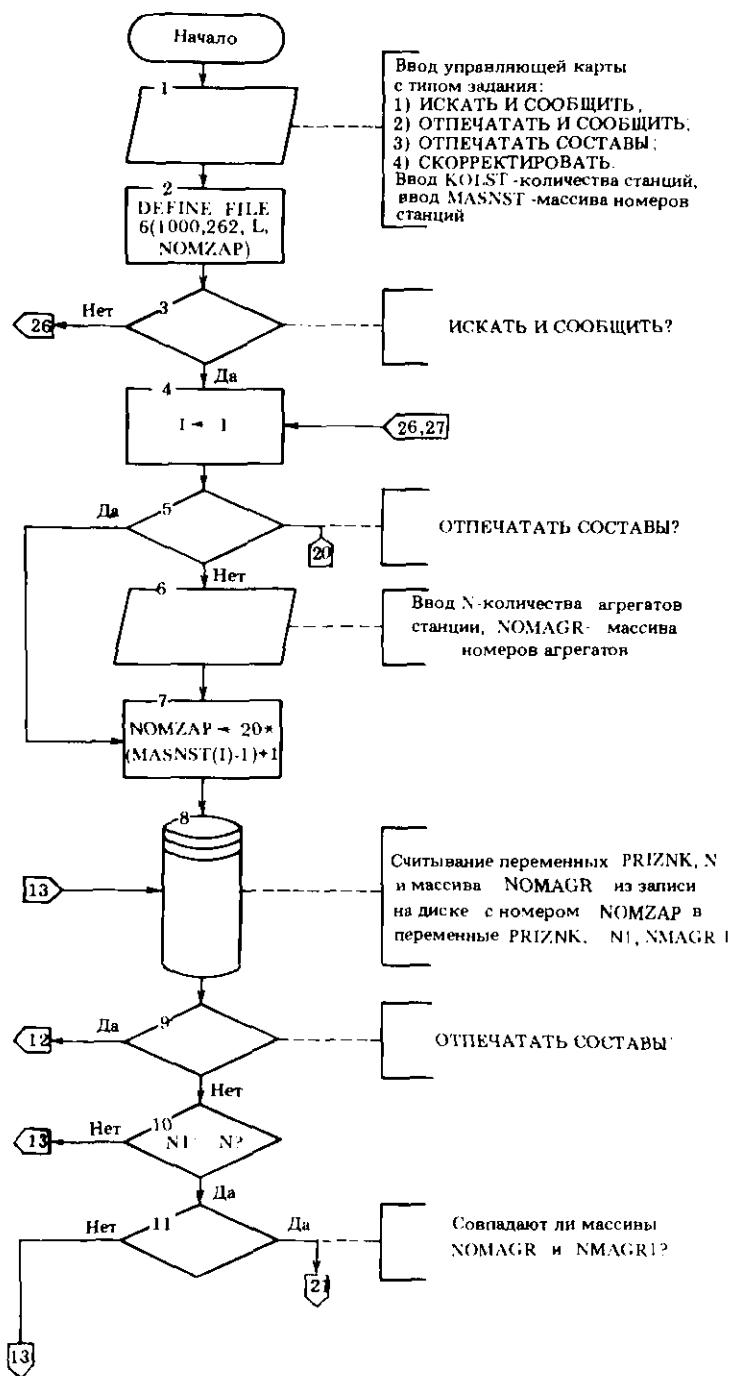


Рис. 24 (начало).

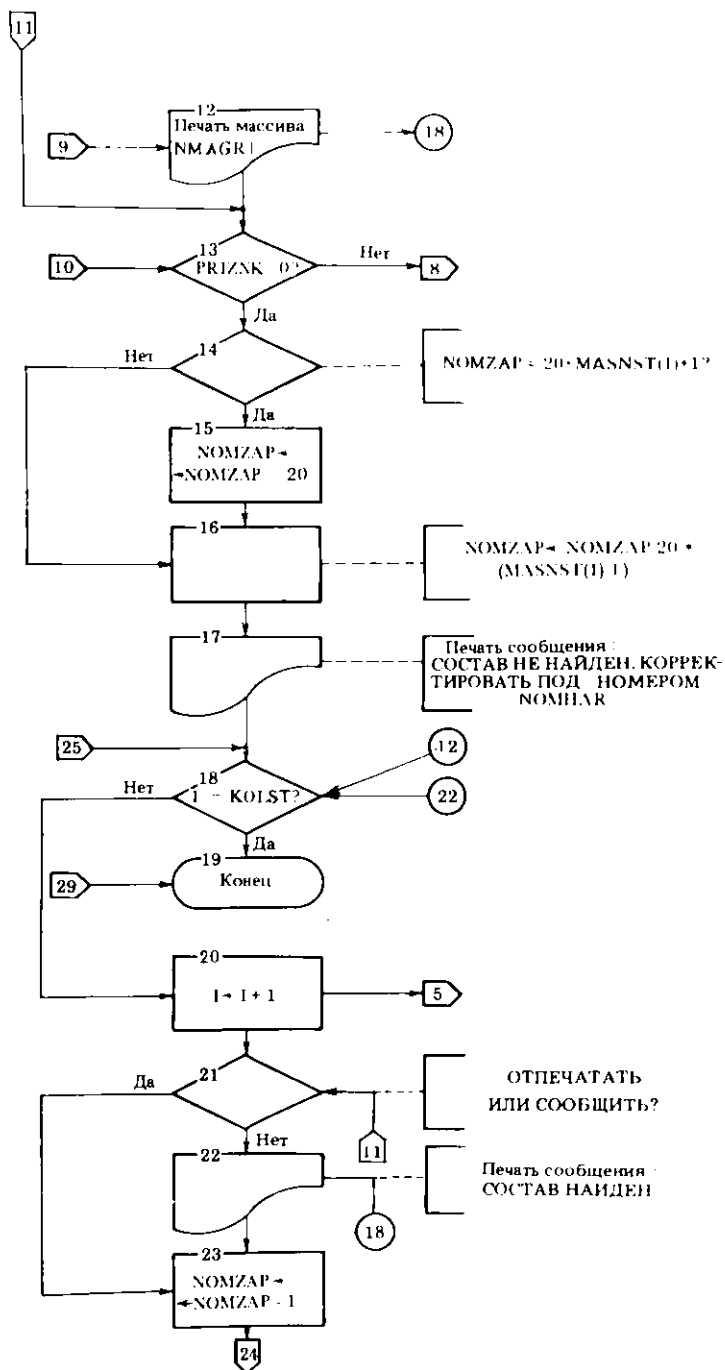


Рис. 24 (продолжение).

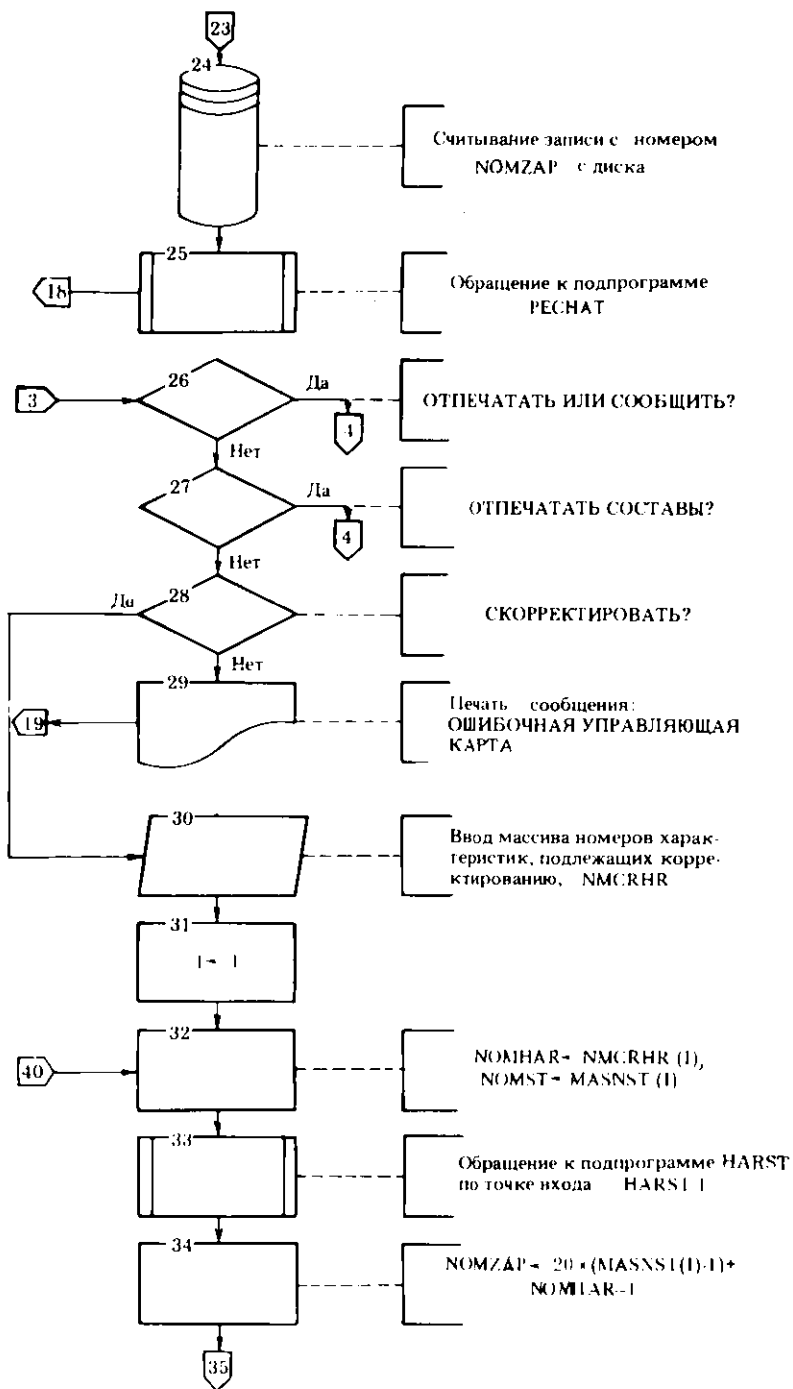


Рис. 24 (окончание).

Если для некоторой станции нужна расходная характеристика имеется в файле, вместе с порядковым номером этой станции печатается сообщение СОСТАВ НАЙДЕН. Когда же требуемая характеристика отсутствует в файле, выдается сообщение СОСТАВ НЕ НАЙДЕН, КОРРЕКТИРОВАТЬ ПОД НОМЕРОМ $NOMHAR=X$, где X — номер подлежащей построению и записи в файле расходной характеристики станции для заданного состава агрегатов в порядке следования характеристик данной станции в соответствующей области файла. Если при этом указанная область не была полностью заполнена характеристиками станции, то сообщенный пользователю номер X соответствует пополнению файла новой характеристикой, в противном случае будет указан номер $X = 1$, что соответствует замещению первой по расположению в файле характеристики данной станции;

2) произвести печать расходных характеристик для заданных составов оборудования, если они имеются в файле, или получить сообщение об отсутствии характеристик и об условии коррекции файла, как описано выше;

3) получить информацию обо всех расходных характеристиках, имеющихся в файле, в виде распечатки наборов эксплуатационных номеров агрегатов, образующих составы включенных в работу агрегатов, которым соответствуют имеющиеся в файле характеристики;

4) построить расходные характеристики, соответствующие заданным составам оборудования, и записать их в нужные области файла для тех станций, для которых предшествующий поиск требуемых характеристик в файле оказался безуспешным.

Контрольный пример

Для иллюстрации возможностей программы CORHAR, форм ее входной и выходной информации, а также с целью наглядного ознакомления с кодированием исходных данных по задаче рассматривается решение контрольного примера. Образцы кодирования данных контрольного примера приведены в графах "Содержимое в контрольном примере" таблиц, представляющих форматы карт управления и данных программы CORHAR. Полностью исходные данные контрольного примера даны в параграфе "Ввод".

Ввод

Режим 1: ИСКАТЬ И СООБЩИТЬ

Карты управления. Набор карт управления включает карту настройки на режим основной программы CORHAR КННРС (табл. 19), карту количества станций ККС (табл. 20) и карты количеств включенных в работу агрегатов ККА1, ККА2, ..., ККА/ для каждой из станций заданного подмножества (в табл. 21 показана карта ККА1 для первой станции).

Карты данных. Набор карт данных включает в себя карту номеров станций КНС (табл. 22) и карты номеров агрегатов КНА1, КНА2, ..., КНА/ для каждой из станций (в табл. 23 показана карта КНА1).

Колода перфокарт. Изображение колоды перфокарт управления и данных приведено на рис. 25. На рис. 26 представлена распечатка вводимых карт для контрольного примера.

Таблица 19

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-24	Требуемый режим	ИСКАТЬ И_СООБЩИТЬ
25-79 80	Пробелы Порядковый номер перфокарты в колоде	1

Таблица 20

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Количество станций	3
4-79 80	Пробелы Порядковый номер перфокарты	2

Таблица 21

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3	Количество агрегатов	7
4-79 80	Пробелы Порядковый номер перфокарты в колоде	4

Таблица 22

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-3/ (3/ + 1) 79 80	Номера станций Пробелы Порядковый номер перфокарты в колоде	3 1 2 3

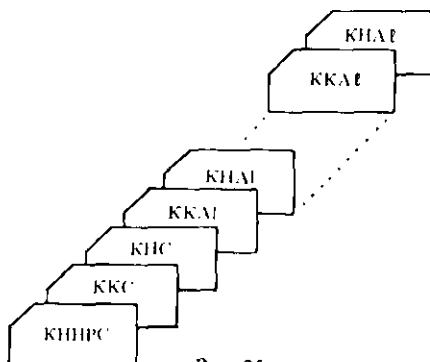


Рис. 25.

```

// JOB РАСПЧТКА
// ASSIGN SYS004.X'00C'
// ASSIGN SYS005.X'00F'
// EXEC CDDP

// UCP TO.FF,A=(00,00),B=(120),Q=(77,4)
// M1
// END
      РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ

CARD TO PRINTER / PUNCH UTILITY
INPUT RECORD LENGTH 0000
OUTPUT RECORD LENGTH 0120
INPUT BLOCK LENGTH 00000
OUTPUT BLOCK LENGTH 00120
INPUT OPTION CARD 00
OUTPUT OPTION PRINT CHARACTER
2 INPUT,2 OUTPUT AREAS ASSIGNED
RECORD FORMAT FIXED
TYPE DATA DISPLAY
STARTING SEQUENCE COLUMN 77
SEQUENCE LENGTH 04
STARTING RECORD NUMBER 00000001
      РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ
      1 ИСКАТЬ И СООБЩИТЬ
BL SZ  BL NO  RC NO.....5.....1.....5.....2.....5.....3.....5.....4.....5.....5.....5.....6.....5.....5.....7.....5.....8.....5.....
      1      2      3      3      3      3      1      2
      4      4      7
      5      5      5      3      1      2      4      6      7
      6      6      12
      7      7      2      5      1      7      4      3      9      6      11      10      8      12
      8      8      10
      9      9      3      1      7      2      5      9      8      4      6      10

END OF DATA
PAGE 1BL NO  RC NO.....5.....1.....5.....2.....5.....3.....5.....4.....5.....5.....6.....5.....7.....5.....8.....5.....

NUMBER OF INPUT BLOCKS PROCESSED 00000009
NUMBER OF OUTPUT BLOCKS PROCESSED 00000009
END OF JOB
EOJ РАСПЧТКА    РАСПЕЧАТКА ВЫПОЛНЕНА

```

Таблица 23

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1 – 3n ₁ (3n ₁ + 1) – 79	Номера агрегатов Пробелы	5 3 1 2 4 6 7
80	Порядковый номер перфокарты в колоде	5

Режим 2: ОТПЕЧАТАТЬ СОСТАВЫ

Карты управления. Требуется две карты управления КННРС и ККС, вид первой из которых показан в табл. 24, а вид второй в контрольном примере такой же, как в табл. 20.

Карты данных. Требуется только одна карта данных КНС, вид которой в контрольном примере такой же, как в табл. 22.

Колода перфокарт. Вид колоды перфокарт управления и данных показан на рис. 27.

Таблица 24

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1 – 24	Требуемый режим	ОТПЕЧАТАТЬ ..
25 – 79	Пробелы	СОСТАВЫ
80	Порядковый номер перфокарты в колоде	1

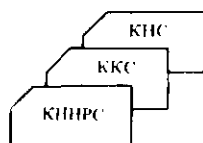


Рис. 27.

Режим 3: ОТПЕЧАТАТЬ ИЛИ СООБЩИТЬ

Состав карт управления и данных и размещение их в колоде такое же, как и для режима ИСКАТЬ И СООБЩИТЬ. Вид карты КННРС для рассматриваемого режима показан в табл. 25.

Таблица 25

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1 – 24	Требуемый режим	ОТПЕЧАТАТЬ, ИЛИ, СООБЩИТЬ ..
25 – 79	Пробелы	
80	Порядковый номер перфокарты в колоде	1

Режим 4: СКОРРЕКТИРОВАТЬ

Карты управления. Набор карт управления включает карты КННПС и ККС, а также необходимые карты управления для подпрограммы HARST, к которой происходит обращение в данном режиме работы.

Вид карты КННПС показан в табл. 26. Карты ККС, ККА1 в контрольном примере имеют такой же вид, как и в табл. 20, 21 соответственно.

Для подпрограммы HARST необходимы карты управления КННРН, КЕИ, КТТ, вид которых в контрольном примере показан соответственно в табл. 27, 28, 29, а также карты ККА1, ККА2, ..., ККА1, которые должны предшествовать наборам карт данных для станций; содержимое карты ККА1 в контрольном примере такое же, как в табл. 21.

Примечания. 1. В табл. 30, показан другой допустимый вид карты КННРН.

2. Смысл обозначений в графе "Характеристика содержимого" табл. 29 (см. в лабораторной работе 1).

Таблица 26

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-24	Требуемый режим	СКОРРЕКТИРОВАТЬ
25-79	Пробелы	
80	Порядковый номер перфокарты в колоде	1

Таблица 27

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-38	Требуемый режим	ОТПЕЧАТАТЬ И ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ
39-79	Пробелы	
80	Порядковый номер перфокарты в колоде	7

Таблица 28

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-16	Единица измерения расхода топлива	T.U.T.
17-79	Пробелы	
80	Порядковый номер перфокарты в колоде	6

Таблица 29

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-9	ΔY^*	0 0 0 0 1.E00
10-18	$\Delta Y_{\text{тах доп}}$	0 0 0 0 5.E00
19-27	ΔP^*	0 0 0 0 2.E00
28-36	$\Delta P^T_{\text{тах доп}}$	0 0 0 0 2.E00
37-79	Пробелы	6
80	Порядковый номер перфокарты в колоде	

Таблица 30

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-38	Требуемый режим	ТОЛЬКО, ЗАПОМНИТЬ...
39-79	Пробелы	ХАРАКТЕРИСТИКИ 0 0 0 0 0 0 0 0
80	Порядковый номер перфокарты в колоде	7

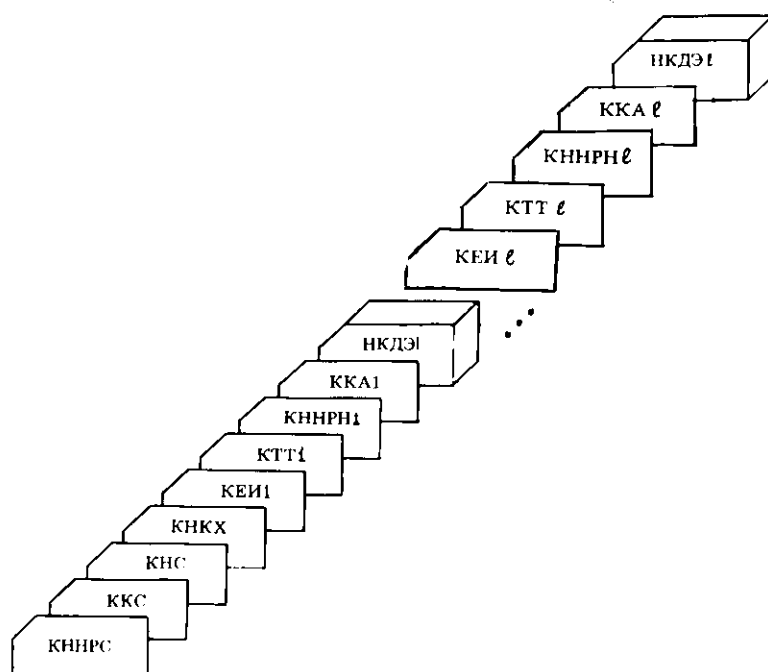


Рис. 28.

Колонка перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1-31	Номера корректируемых характеристик станций в областях файла	2 2 2 2
(31 + 1) - 79	Пробелы	
80	Порядковый номер перфокарты в колоде	4

Карты данных. Набор карт данных включает карту КНС, вид которой в контрольном примере такой же, как в табл. 22, карту КНКХ номеров корректируемых характеристик станций (табл. 31) и наборы карт данных электростанций НКДЭ1, НКДЭ2, ..., НКДЭ1, которые подготавливаются, как описано в лабораторной работе 1.

Колода перфокарт. Изображение колоды перфокарт управления и данных приведено на рис. 28. На рис. 29 представлена распечатка вводимых карт для контрольного примера.

Вывод

Состав информации, выводимой на печать в каждом из возможных режимов работы программы, охарактеризован в подразделе "Возможности". Печать результатов контрольного примера для всех возможных режимов работы программы приведена соответственно на рис. 30-33.

Инструкция по применению программы

Программа CORHAR каталогизирована в личную библиотеку абсолютных модулей. Задание на исполнение программы может иметь, например, следующий вид (соответствует примеру задания на работу программы POSHARST, приведенному в лабораторной работе 3).

```
// JOB КОРФАЙЛА  КОРРЕКЦИЯ_ ФАЙЛА _ РАСХОДНЫХ _
  ХАРАКТЕРИСТИК'
ASSGN SYSCLB,X'191'
// ASSGN SYS003,X'191'
// DLBL IISYS03,'ФАЙЛ_РАСХОДНЫХ_ХАРАКТЕРИСТИК_
  СТАНЦИЙ'
// EXTENT,,111111,,11,100
// EXEC CORHAR
  (Файл карт управления и данных)
/*
ASSGN SYSCLB,UA
/& РЕШЕНИЕ_ЗАДАЧИ_ПО_
  ВАРИАНТУ_№_ВЫПОЛНЕНО
```

Примечание. В позицию 42 карты /& заносится номер указанного преподавателем варианта задания на выполнение лабораторной работы.

```

// JOB ПАСЧУКА
// ASSEM SYS004,X'00FC'
// ASSEM SYS005,X'00FF'
// EXEC CDDP

// UCD TD,FF,A=(00,80),B=(120),Q=(77,4)
// M1
// END
// ПАСЧЕЧАТКА КАРТ УПРАВЛЕНИЯ И ААННЫХ

CARD TO PRINTER / PUNCH UTILITY
INPUT RECORD LENGTH 0000
OUTPUT RECORD LENGTH 0120
INPUT BLOCK LENGTH 0000
OUTPUT BLOCK LENGTH 0012
INPUT OPTION CARD BCD
OUTPUT OPTION PRINT CHARACTER
2 INPUT, 2 OUTPUT AREAS ASSIGNED
RECORD FORMAT FIXED
TYPE DATA DISPLAY
STARTING SEQUENCE COLUMN 77
SEQUENCE LENGTH 04
STARTING RECORD NUMBER 00000001

BL SZ BL NO AC NO.....5.....1.....5.....2.....5.....3.....5.....4.....5.....5.....7.....5.....8.....5.....
1 1 1 СКОРРЕКТИРОВАТЬ
2 2 3
3 3 1 2 3
4 4 2 2 2
5 5 T W.T.
6 6 1.000 5.000 2.000 2.000
7 7 ОТПЕЧАТАТЬ И ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ
8 8 10
9 9 1 4 2 4 3 4 4 4 5 5 6 4 7 4 8 4 9 4 10 4 2.000
10 10 10.00 2.000 12.00 2.000 14.00 2.000
11 11 17.00 4.000 20.00 5.000
12 12 40.00 8.000 50.00 9.500
13 13 70.00 10.000 80.00 11.000
14 14 30.00 6.000 40.00 8.500
15 15 65.00 20.00 75.00 26.00

```

Рис. 29 (начало).

16	25.ЕВ	5.ЕВ	3В.ЕВ	5.5ЕВ	4В.ЕВ	6.5ЕВ	16
17	5В.ЕВ	1В.5ЕВ	6В.ЕВ	14.5ЕВ			17
18	3В.ЕВ	6.ЕВ	4В.ЕВ	7.3ЕВ	5В.ЕВ	8.6ЕВ	18
19	6В.ЕВ	13.9ЕВ	7В.ЕВ	19.2ЕВ	8В.ЕВ	24.5ЕВ	19
2В	4В.ЕВ	8.ЕВ	5В.ЕВ	9.5ЕВ	7В.ЕВ	12.5ЕВ	2В
21	8В.ЕВ	17.ЕВ	9В.ЕВ	21.5ЕВ			21
22	2В.ЕВ	4.ЕВ	3В.ЕВ	5.5ЕВ	35.ЕВ	6.25ЕВ	22
23	45.ЕВ	10.25ЕВ	5В.ЕВ	12.25ЕВ			23
24	35.ЕВ	7.ЕВ	45.ЕВ	9.ЕВ	55.ЕВ	11.ЕВ	24
25	7В.ЕВ	17.75ЕВ	8В.ЕВ	22.25ЕВ			25
26	5В.ЕВ	1В.ЕВ	6В.ЕВ	11.7ЕВ	7В.ЕВ	13.4ЕВ	26
27	8В.ЕВ	16.9ЕВ	1ВВ.ЕВ	25.9ЕВ			27
28	4В.ЕВ	8.ЕВ	5В.ЕВ	9.8ЕВ	6В.ЕВ	11.6ЕВ	28
29	7В.ЕВ	15.6ЕВ	9В.ЕВ	23.6ЕВ			29
3В	7 В.Т.						3В
31	1.ЕВ	5.ЕВ	2.ЕВ	2.ЕВ			31
32	ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКУ						32
33	12						33
34	1 4 2 4 3 4 4 4 5 5 6 4 7 4 8 4 9 4 10 5 12 6						34
35	1В.ЕВ	2.ЕВ	12.ЕВ	2.4ВВ	14.ЕВ	2.8ЕВ	35
36	17.ЕВ	4.3ЕВ	2В.ЕВ	5.8ЕВ			36
37	4В.ЕВ	8.ЕВ	5В.ЕВ	9.5ЕВ	6В.ЕВ	11.ЕВ	37
38	7В.ЕВ	16.5ЕВ	6В.ЕВ	22.ЕВ			38
39	3В.ЕВ	6.ЕВ	4В.ЕВ	8.5ЕВ	5В.ЕВ	11.ЕВ	39
4В	65.ЕВ	2В.ЕВ	75.ЕВ	26.ЕВ			4В
41	25.ЕВ	5.ЕВ	3В.ЕВ	5.5ЕВ	4В.ЕВ	6.5ЕВ	41
42	5В.ЕВ	1В.5ЕВ	6В.ЕВ	14.5ЕВ			42
43	3В.ЕВ	6.ЕВ	4В.ЕВ	7.3ЕВ	5В.ЕВ	8.6ЕВ	43
44	6В.ЕВ	13.9ЕВ	7В.ЕВ	19.2ЕВ	8В.ЕВ	24.5ЕВ	44
45	4В.ЕВ	8.ЕВ	5В.ЕВ	9.5ЕВ	7В.ЕВ	12.5ЕВ	45
46	8В.ЕВ	17.ЕВ	9В.ЕВ	21.5ЕВ			46
47	2В.ЕВ	4.ЕВ	3В.ЕВ	5.5ЕВ	35.ЕВ	6.25ЕВ	47
48	45.ЕВ	1В.25ЕВ	5В.ЕВ	12.25ЕВ			48
49	35.ЕВ	7.ЕВ	45.ЕВ	9.ЕВ	55.ЕВ	11.ЕВ	49
5В	7В.ЕВ	17.75ЕВ	8В.ЕВ	22.25ЕВ			5В
51	5В.ЕВ	10.ЕВ	6В.ЕВ	11.7ЕВ	7В.ЕВ	13.4ЕВ	51
52	8В.ЕВ	16.9ЕВ	1ВВ.ЕВ	23.9ЕВ			52
53	4В.ЕВ	8.ЕВ	5В.ЕВ	9.8ЕВ	6В.ЕВ	11.6ЕВ	53
54	7В.ЕВ	15.6ЕВ	9В.ЕВ	23.6ЕВ			54
55	5В.ЕВ	1В.ЕВ	6В.ЕВ	11.5ЕВ	7В.ЕВ	13.ЕВ	55

Рис. 29 (продолжение).

56	56	89.ЕВ	16.ЕВ	9В.ЕВ	19.ЕВ	1.Е2	23.5ЕВ	56
57	57	3В.ЕВ	6.ЕВ	4В.ЕВ	8.ЕВ	45.ЕВ	9.ЕВ	57
58	58	5В.ЕВ	11.ЕВ	6В.ЕВ	15.ЕВ	7В.ЕВ	21.ЕВ	58
59	59	75.ЕВ	24.ЕВ					59
60	60	7 У.Т.						60
61	61	1.ЕВ	5.ЕВ	2.ЕВ	2.ЕВ			61
62	62	ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ						62
63	63							63
64	64	1 4 2 3 4 4 4 5 6 4 7 4 8 4 9 4						64
РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ								
BL SZ	BL NO	5	1	5	2	5	3	5
65	65	1В.ЕВ	2.ЕВ	12.ЕВ	2.4ЕВ	14.ЕВ	2.8ЕВ	65
66	66	17.ЕВ	4.3ЕВ	2В.ЕВ	5.8ЕВ			66
67	67	4В.ЕВ	8.ЕВ	5В.ЕВ	9.5ЕВ	60.ЕВ	11.ЕВ	67
68	68	7В.ЕВ	16.5ЕВ	8В.ЕВ	22.ЕВ			68
69	69	3В.ЕВ	6.ЕВ	4В.ЕВ	8.5ЕВ	50.ЕВ	11.ЕВ	69
70	70	65.ЕВ	2В.ЕВ	75.ЕВ	26.ЕВ			70
71	71	25.ЕВ	5.ЕВ	3В.ЕВ	5.5ЕВ	40.ЕВ	6.5ЕВ	71
72	72	5В.ЕВ	10.5ЕВ	6В.ЕВ	14.5ЕВ			72
73	73	3В.ЕВ	6.ЕВ	4В.ЕВ	7.3ЕВ	50.ЕВ	8.6ЕВ	73
74	74	4В.ЕВ	13.9ЕВ	7В.ЕВ	19.2ЕВ	80.ЕВ	24.5ЕВ	74
75	75	4В.ЕВ	8.ЕВ	5В.ЕВ	9.5ЕВ	70.ЕВ	12.5ЕВ	75
76	76	8В.ЕВ	17.ЕВ	9В.ЕВ	21.5ЕВ			76
77	77	2В.ЕВ	4.ЕВ	3В.ЕВ	5.5ЕВ	35.ЕВ	6.25ЕВ	77
78	78	45.ЕВ	10.25ЕВ	5В.ЕВ	12.25ЕВ			78
79	79	35.ЕВ	7.ЕВ	45.ЕВ	9.ЕВ	55.ЕВ	11.ЕВ	79
80	80	7В.ЕВ	17.75ЕВ	8В.ЕВ	22.25ЕВ			80
81	81	5В.ЕВ	1В.ЕВ	6В.ЕВ	11.75ЕВ	7В.ЕВ	13.4ЕВ	81
82	82	8В.ЕВ	16.9ЕВ	10В.ЕВ	23.9ЕВ			82
END OF DATA								
PAGE	2BL NO	5	1	5	2	5	3	5
NUMBER OF INPUT BLOCKS PROCESSED	82							
NUMBER OF OUTPUT BLOCKS PROCESSED	82							
END OF JOB								
END RASCHETKA	РАСПЕЧАТКА ВЫПОЛНЕНА							

12.36.01.DURATION 08.08.19

Рис. 29 (окончание).

// JOB КОФААЛА 'КОРРЕКЦИЯ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК'

11.31.48

// DLBL IJSYS03,'FILE RASHARST'

// EXTENT SYS003,111111,.,11,100

// ASSIGN SYS003,X'191'

// EXEC CORHAR

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 3 НАРДЕН ПОД НОМЕРОМ 1

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 1 НАРДЕН ПОД НОМЕРОМ 1

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 2 НАРДЕН ПОД НОМЕРОМ 1

ЕОJ КОФААЛА РЕШЕНИЕ ЗАДАЧИ ПО ВАРИАНТУ N0 ВЫПОЛНЕНО

11.32.10,DURATION 00.00.22

Рис. 30.

// JOB КОФААЛА 'КОРРЕКЦИЯ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК'

// DLBL IJSYS03,'FILE RASHARST',10,SD

// EXTENT SYS003,111111,.,11,100

// ASSIGN SYS003,X'191'

// EXEC CORHAR

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 2

АГРЕГАТЫ NN: 1, 2, 3, 4, 5, 6, 7, 8, 9,10,

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 1

АГРЕГАТЫ NN: 1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12,

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N 3

АГРЕГАТЫ NN: 1, 2, 3, 4, 5, 6, 7,

ЕОJ КОФААЛА РЕШЕНИЕ ЗАДАЧИ ПО ВАРИАНТУ N0 ВЫПОЛНЕНО 11.36.48,DURATION 00.00.21

Рис. 31.

```
// JOB: КОРЕКЦИЯ ФАЙЛА РАСХОДНЫХ ХАРАКТЕРИСТИК'
// DLBL IJSD3, FILE RASHARST', 1B, SD
// EXTENT SYSDB3, 111111, ..., 11, 100
// ASSGN SYSDB3, X'101'
// EXEC CORNAR
```

СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ № 4 ИЕ НАДЕН, КОРРЕКТИРОВАТЬ ПОД НОМЕРОМ 2

РАСХОДНАЯ ХАРАКТЕРИСТИКА СТАНЦИИ № 1

ВКЛЮЧЕНЫ В РАБОТУ АГРЕГАТЫ МН: 1 2 3 4 5 6 7 8 9 10 11 12

Нагрузка / станция	1	2	3	4	5	6	7	8	9	10	11	12
Мвт	18.408Е	131В.411Е	131В.422Е	131В.433Е	131В.444Е	131В.466Е	131В.477Е	131В.488Е	131В.499Е	131В.510Е	131В.521Е	131В.532Е
Расход топлива	131В.543Е	131В.554Е	131В.565Е	131В.576Е	131В.587Е	131В.598Е	131В.609Е	131В.620Е	131В.631Е	131В.642Е	131В.653Е	131В.664Е
Т.У.Т.	131В.675Е	131В.686Е	131В.697Е	131В.708Е	131В.719Е	131В.730Е	131В.741Е	131В.752Е	131В.763Е	131В.774Е	131В.785Е	131В.796Е
Нагрузка / станция	1	2	3	4	5	6	7	8	9	10	11	12
Мвт	18.408Е	131В.411Е	131В.422Е	131В.433Е	131В.444Е	131В.466Е	131В.477Е	131В.488Е	131В.499Е	131В.510Е	131В.521Е	131В.532Е
Расход топлива	131В.543Е	131В.554Е	131В.565Е	131В.576Е	131В.587Е	131В.598Е	131В.609Е	131В.620Е	131В.631Е	131В.642Е	131В.653Е	131В.664Е
Т.У.Т.	131В.675Е	131В.686Е	131В.697Е	131В.708Е	131В.719Е	131В.730Е	131В.741Е	131В.752Е	131В.763Е	131В.774Е	131В.785Е	131В.796Е
Нагрузка / станция	1	2	3	4	5	6	7	8	9	10	11	12
Мвт	18.408Е	131В.411Е	131В.422Е	131В.433Е	131В.444Е	131В.466Е	131В.477Е	131В.488Е	131В.499Е	131В.510Е	131В.521Е	131В.532Е
Расход топлива	131В.543Е	131В.554Е	131В.565Е	131В.576Е	131В.587Е	131В.598Е	131В.609Е	131В.620Е	131В.631Е	131В.642Е	131В.653Е	131В.664Е
Т.У.Т.	131В.675Е	131В.686Е	131В.697Е	131В.708Е	131В.719Е	131В.730Е	131В.741Е	131В.752Е	131В.763Е	131В.774Е	131В.785Е	131В.796Е

Рис. 32 (начало).

Сообщения об ошибках и время решения

Во время исполнения программы, кроме стандартных сообщений системы, может быть выдано сообщение проблемной программы ОШИБОЧНАЯ УПРАВЛЯЮЩАЯ КАРТА, если информация на управляющей карте КННРС или расположение символов на ней не соответствуют ни одному из режимов, предусмотренных программой.

Время решения контрольного примера для каждого из ее возможных режимов составляет соответственно 22 с, 21 с, 43 с, 1 мин. 57 с.

Подпрограммы

Все подпрограммы, содержащиеся в программе CORHAR, описаны в лабораторной работе 3.

3. ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

В данной лабораторной работе ставится задача приобретения навыков обслуживания файлов прямого доступа, используемых в задачах АСУ, с помощью специальных программ, служащих для этой цели, на примере программы CORHAR. Эта программа осуществляет коррекцию файла расходных характеристик электростанций, необходимого для задачи оптимального распределения активных мощностей в энергосистеме с использованием декомпозиции задачи на подзадачи внутри- и межстанционной оптимизации. Соответственно при подготовке к выполнению лабораторной работы необходимо:

1) убедиться в достаточной степени усвоения материала ранее выполненной лабораторной работы 3;

2) изучить раздел 1 данной лабораторной работы, обратив особое внимание на характер взаимодействия программ POSHARST и CORHAR при использовании файла расходных характеристик в задаче оптимизации режима энергосистемы по активной мощности;

3) изучить раздел 2, уделив особое внимание вопросам подготовки колоды перфокарт управления и данных в различных режимах работы программы CORHAR;

4) получить у преподавателя один из вариантов задания на выполнение лабораторной работы;

5) подготовить исходную информацию по предложенному варианту задания на бланках для сдачи на перфорацию.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать в себя следующие разделы:

- 1) цель работы;
- 2) описание режимов работы программы CORHAR;
- 3) схему программы CORHAR;
- 4) изображение колоды перфокарт управления и данных для заданного варианта исходных данных;

- 5) задание операционной системе для соответствующего варианта исходных данных;
- 6) сообщения операционной системы о результатах обработки задания;
- 7) анализ результатов обработки задания;
- 8) график расходной характеристики одной из станций.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Для чего необходимо накопление ретроспективных данных в составе базы УПИ для задач АСУ энергосистемой?
2. По какому принципу строится файл расходных характеристик электростанций программами POSHARST и CORHAR?
3. Как производится пополнение и обновление файла расходных характеристик станций программой CORHAR?
4. Почему для хранения расходных характеристик станций энергосистемы используется файл прямого, а не последовательного доступа?
5. Охарактеризуйте функции, выполняемые программой CORHAR в каждом из возможных режимов ее работы?
6. Для чего в составе результатов работы программы в режимах ИСКАТЬ И СООБЩИТЬ и ОТПЕЧАТАТЬ ИЛИ СООБЩИТЬ выдается сообщение КОРРЕКТИРОВАТЬ ПОД НОМЕРОМ NOMHAR = X?
7. Как зависит режим работы программы CORHAR от значения карты настройки на режим подпрограммы HARST?

6. ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Вариант 1

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-7	$\frac{1}{5}$	$\frac{3}{5}$	$\frac{5}{4}$	$\frac{7}{4}$		
ТЭЦ-3*	$\frac{2}{1}$	$\frac{4}{1}$	$\frac{7}{3}$	$\frac{3}{1}$	$\frac{5}{3}$	$\frac{8}{1}$

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,5\%$; $\Delta Y_{\text{max доп}} = 5$ МВт. Режимы: 1) ОТПЕЧАТАТЬ ИЛИ СООБЩИТЬ; 2) СКОРРЕКТИРОВАТЬ без вывода характеристики на печать.

Вариант 2

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)				
ТЭЦ-2*	$\frac{2}{2}$	$\frac{1}{2}$	$\frac{5}{1}$	$\frac{4}{2}$	$\frac{8}{3}$
ГРЭС-2	$\frac{2}{4}$	$\frac{7}{6}$	$\frac{4}{4}$	$\frac{6}{4}$	

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\text{max доп}} = 9$ МВт. Режимы: 1) ОТПЕЧАТАТЬ ИЛИ СООБЩИТЬ; 2) СКОРРЕКТИРОВАТЬ без вывода характеристики на печать.

Вариант 3

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)				
ТЭЦ-2	$\frac{6}{1}$	$\frac{2}{2}$	$\frac{5}{1}$	$\frac{7}{1}$	$\frac{1}{2}$
ГРЭС-3*	$\frac{1}{4}$	$\frac{5}{5}$	$\frac{7}{5}$	$\frac{2}{4}$	

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,6\%$; $\Delta Y_{\text{мах доп}} = 10$ МВт. Режимы: 1) ИСКАТЬ И СООБЩИТЬ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

Вариант 4

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-5*	$\frac{6}{5}$	$\frac{4}{4}$	$\frac{7}{5}$	$\frac{5}{4}$		
ТЭЦ-5	$\frac{2}{3}$	$\frac{5}{3}$	$\frac{11}{1}$	$\frac{7}{1}$	$\frac{9}{1}$	$\frac{10}{1}$

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,7\%$; $\Delta Y_{\text{мах доп}} = 7$ МВт. Режимы: 1) ИСКАТЬ И СООБЩИТЬ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

Вариант 5

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)					
ГРЭС-8	$\frac{5}{1}$	$\frac{9}{4}$	$\frac{7}{4}$	$\frac{4}{1}$	$\frac{3}{1}$	
ТЭЦ-4*	$\frac{1}{2}$	$\frac{9}{3}$	$\frac{8}{3}$	$\frac{2}{2}$	$\frac{4}{1}$	$\frac{6}{1}$

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,6\%$; $\Delta Y_{\text{мах доп}} = 8$ МВт. Режимы: 1) ИСКАТЬ И СООБЩИТЬ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

Вариант 6

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)				
ГРЭС-2*	$\frac{5}{6}$	$\frac{4}{4}$	$\frac{1}{5}$		
ГРЭС-9	$\frac{2}{4}$	$\frac{8}{6}$	$\frac{4}{4}$	$\frac{6}{6}$	

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,5\%$; $\Delta Y_{\text{мах доп}} = 5$ МВт. Режимы: 1) ОТПЕЧАТАТЬ СОСТАВЫ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)						
ТЭЦ-1	$\frac{2}{2}$	$\frac{4}{2}$	$\frac{10}{3}$	$\frac{11}{3}$	$\frac{7}{3}$	$\frac{3}{2}$	$\frac{5}{3}$
ГРЭС-3*	$\frac{11}{1}$	$\frac{7}{3}$	$\frac{3}{2}$	$\frac{4}{2}$	$\frac{5}{2}$		

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,7\%$; $\Delta Y_{\text{max доп}} = 2,5 \text{ МВт}$. Режимы: 1) ОТПЕЧАТАТЬ СОСТАВЫ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов (в числителе) и типы их расходных характеристик (в знаменателе)			
ГРЭС-7	$\frac{1}{5}$	$\frac{5}{4}$	$\frac{3}{4}$	$\frac{4}{4}$
ГРЭС-4*	$\frac{1}{5}$	$\frac{7}{6}$	$\frac{2}{5}$	$\frac{4}{5}$

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,6\%$; $\Delta Y_{\text{max доп}} = 4,5 \text{ МВт}$. Режимы: 1) ОТПЕЧАТАТЬ СОСТАВЫ; 2) СКОРРЕКТИРОВАТЬ с выводом характеристики на печать.

П р и м е ч а н и я. 1. Типы расходных характеристик агрегатов, подразумеваемые в вариантах заданий, приводятся в табл. 18.

2. По каждому варианту необходимо подготовить задание для операционной системы, состоящее из двух шагов, на каждом из которых должна выполняться программа CORHAR, настроенная соответственно на режимы 1 и 2, указанные в вариантах исходных данных.

3. При составлении заданий необходимо учитывать структуру файла, созданного с помощью программы POSHARST при выполнении лабораторной работы 3 по варианту исходных данных с тем же номером, что и в настоящей работе. Для этого необходимо иметь при себе пакет составленного ранее шага задания на исполнение программы POSHARST

4. Для выполнения программы CORHAR в режиме СКОРРЕКТИРОВАТЬ колода карт управления и данных должна быть подготовлена только для станции, наименование которой отмечено символом * в таблице исходных данных по соответствующему варианту.

5. Следует сохранить у себя текст шага задания на исполнение программы POSHARST из лабораторной работы 3, так как он понадобится при выполнении лабораторной работы 5.

**ПОДГОТОВКА КОМПЛЕКСА ПРОГРАММ ОПЕРАТИВНОЙ
ОПТИМИЗАЦИИ РЕЖИМА ЭНЕРГОСИСТЕМЫ ПО АКТИВНОЙ
МОЩНОСТИ К ЭКСПЛУАТАЦИИ И ОПТИМАЛЬНОЕ
РАСПРЕДЕЛЕНИЕ МОЩНОСТЕЙ МЕЖДУ ЭЛЕКТРОСТАНЦИЯМИ**

Ц е л ь р а б о т ы: изучить на примере задачи оптимизации режима энергосистемы по активной мощности вопросы взаимоувязки набора программных модулей в комплекс подготовленных к эксплуатации программ решения подзадач, входящих в состав данной задачи; получить навыки эксплуатации программ решения подзадач, использующих базу условно-постоянной информации, на примере подзадачи оптимального распределения активных мощностей между электростанциями.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

**Подготовка комплекса программ решения подзадач,
входящих в состав единой задачи, на основе набора исходных
программных модулей на языке ФОРТРАН IV**

При разработке программного обеспечения задачи, исходя из состава входящих в нее подзадач и их алгоритмической структуры, выявляется ряд функций по переработке информации, каждая из которых необходима более чем для одной подзадачи.

С целью снижения затрат на программирование каждую из таких функций целесообразно реализовать в виде подпрограммы, которая в дальнейшем должна быть включена в программы решения соответствующих подзадач. По мере составления и отладки необходимых программных модулей (или их подбора из числа опубликованных) они каталогизируются программистом в личные или системную библиотеки исходных модулей (см. лабораторную работу 2).

При программировании на языке ФОРТРАН IV для каждой из автономных подзадач должна быть составлена отдельная программа, включающая как обязательную программную единицу основную программу, в которой содержится обращение по крайней мере к одной из подпрограмм из числа требуемых для данной подзадачи (обращения к остальным подпрограммам могут находиться либо в основной программе, либо в подпрограммах). Модули ФОРТРАНА IV, соответствующие основным программам для решения подзадач, также каталогизируются в библиотеки исходных модулей.

Последующие этапы подготовки из наборов библиотечных модулей программ решения автономных подзадач осуществляются операционной системой на основе заданий, которые могут составляться пользователем, обязанным владеть соответствующей методикой.

Если программные модули для данной задачи, каталогизированные в библиотеку исходных модулей, полностью отлажены, то этап формирования из отдельных модулей автономных программ решения каждой подзадачи (редак-

тирование программы) целесообразно совмещать в одном задании с предшествующим этапом трансляции исходных модулей в объектные.

В распределении обязанностей между операционной системой и пользователем при формировании текста исполнимой программы решения подзадачи существенным является следующее обстоятельство. Если обращение к подпрограмме содержится в основной программе, подпрограмма может быть включена в текст программы РЕДАКТОРОМ в стандартном режиме работы операционной системы AUTOLINK. В противном случае пользователь должен позаботиться о подготовке в составе задания необходимых управляющих операторов РЕДАКТОРА (см. лабораторную работу 2).

Заключительным этапом подготовки программ решения подзадач к эксплуатации является каталогизация полученных после редактирования исполнимых программ в системные или личные библиотеки абсолютных модулей, что позволяет в дальнейшем осуществлять многократное оперативное решение подзадач без затрат времени на предварительную подготовку программ к исполнимому виду.

Для работы с личными библиотеками исходных, объектных и абсолютных модулей, созданных программистом, пользователю необходима следующая информация:

1) он должен знать, на каких конкретных пакетах дисков (идентифицируемых регистрационными номерами; см. лабораторную работу 3) размещены соответствующие библиотеки. Пользователь сообщает оператору ЭВМ, какой пакет дисков следует установить на дисководе, соответствующем физическому устройству ввода-вывода, которое назначено пользователем логическому устройству для данной личной библиотеки;

2) ему должно быть известно, какое значение параметра "идентификатор файла" было задано программистом в управляющем операторе DLBL программы УПРАВЛЕНИЕ ЗАДАНИЯМИ (лабораторная работа 3) при создании файла для размещения личной библиотеки. Это необходимо пользователю для правильного составления задания, в котором осуществляется вызов модулей из личной библиотеки или каталогизация в личную библиотеку, в тех случаях, когда программист при создании библиотеки использовал значение параметра, отличное от стандартного (см. лабораторную работу 3). Задание уникального значения параметру "идентификатор файла" является обязательным, если на одном пакете дисков размещается несколько личных библиотек одноименного назначения.

Если при создании библиотеки использовалось стандартное значение параметра, пользователь избавлен от необходимости предусматривать в задании операторы DLBL и EXTENT, сообщающие информацию о метках файла, содержащего личную библиотеку. Когда же для параметра "идентификатор файла" предусмотрено уникальное значение, пользователю достаточно включить в задание оператор DLBL в следующей простейшей форме:

// DLBL имя файла, 'идентификатор файла'

3) он должен знать, какой порядковый номер присвоен программистом подбиблиотеке ФОРТРАНа IV в библиотеке исходных модулей.

Пример 1. Составить задание на трансляцию, редактирование и каталогизацию в личную библиотеку абсолютных модулей программы ФОРТРАНа IV RASNAGA2, находящейся в личной библиотеке исходных модулей в подбиблиотеке с номером 1, если известно,

что файл с библиотекой исходных модулей имеет стандартное значение параметра "идентификатор файла", а файл с библиотекой абсолютных модулей — значение 'БИБЛИОТЕКА_АБСОЛЮТНЫХ_МОДУЛЕЙ', и обе библиотеки размещены на одном пакете дисков.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB TPPEДКАТ
// OPTION CATAL,NOLIST
// ASSGN SYSSLB,X'191'
// ASSGN SYSPCH, X'280'
// EXEC SSERV
  PUNCH1.RASNAGA2
// ASSGN SYSIPT,X'280'
// MTC WTM,X'280'
// MTC BSF,X'280'
  ASSGN SYSCLB,X'191'
// DLBL IJSYSCLB,'БИБЛИОТЕКА_АБСОЛЮТНЫХ_МОДУЛЕЙ'
// EXEC FFORTTRAN
// EXEC LNKEДT
  ASSGN SYSCLB, UA
/&
```

Принятый в выполняемом лабораторном практикуме состав подзадач задачи оптимизации режима энергосистемы по активной мощности определяется упрощающими допущениями при выборе математической модели задачи, методикой декомпозиции задачи, соответствующей иерархической структуре энергосистемы, способом организации базы условно-постоянной информации (см. лабораторные работы 3, 4). Программы решения отдельных подзадач описаны в лабораторных работах 1, 3, 4.

Приведем полный перечень подзадач рассматриваемой задачи с указанием относящихся к ним программных модулей ФОРТРАНа IV и вызывающих модулей для модулей-подпрограмм.

1. Подзадача оптимального распределения активных мощностей между агрегатами электростанций выполняется основной программой RASNAGA1, которая включает подпрограмму DYNPRG, вызываемую модулем RASNAGA1.

2. Подзадача организации файла расходных характеристик электростанций выполняется основной программой POSHARST, которая включает подпрограммы HARST, DYNPRG, PECHAT, вызываемые соответственно модулями POSHARST, HARST, HARST.

3. Подзадача коррекции файла расходных характеристик электростанций решается основной программой CORHAR, которая включает подпрограммы HARST, DYNPRG, PECHAT, вызываемые соответственно модулями CORHAR, HARST, CORHAR и HARST.

4. Подзадача оптимального распределения активных мощностей между электростанциями энергосистемы выполняется основной программой RASNAGST, которая включает подпрограмму DYNPRG, вызываемую модулем RASNAGST.

Все перечисленные модули каталогизированы в личную библиотеку исходных модулей.

2. ПРОГРАММА RASNAGST ОПТИМАЛЬНОГО РАСПРЕДЕЛЕНИЯ АКТИВНЫХ МОЩНОСТЕЙ МЕЖДУ ЭЛЕКТРОСТАНЦИЯМИ ЭНЕРГОСИСТЕМЫ

Описание задачи

Программа служит для решения задачи межстанционной оптимизации активных мощностей:

$$\sum_{i=1}^m \varphi_i(P_i) \rightarrow \min; \quad (20)$$

$$\sum_{i=1}^m P_i = P; \quad (21)$$

$$P_{\min} \leq P_i \leq P_{\max}, i = 1, 2, \dots, m, \quad (22)$$

где P — суммарная активная мощность потребления в энергосистеме с учетом прогноза потерь активной мощности в сети; $P_i, P_{\min}, P_{\max}, i = 1, 2, \dots, m$, — соответственно искомые активные мощности станций энергосистемы, их технические минимумы нагрузки и располагаемые мощности для заданных составов включенных в работу агрегатов; $\varphi_i(P_i), i = 1, 2, \dots, m$, — соответствующие расходные характеристики электростанций, хранящиеся в заблаговременно подготовленном файле расходных характеристик (см. лабораторные работы 3, 4).

Программа решения задачи

Описание. Состав исходных модулей и характер взаимных ссылок между ними описаны в разделе 1 данной лабораторной работы. Схема программы показана на рис. 34. Текст исходного модуля ее приведен в приложении.

Примечание. На рис. 34 (см. блок 14) I — номер станции в порядке расположения относящихся к ней данных в колоде перфокарт; X — номер подлежащей построению и записи в файл расходной характеристики I -й станции в порядке следования ее характеристик в соответствующей области файла.

Возможности. Предельное количество m станций в энергосистеме 50.

Обеспечиваемые программой возможности по предельно достижимой точности решения задачи определяются величинами шагов ΔY и ΔP_{τ} на прямом ходе МДП (см. лабораторную работу 1) для межстанционной задачи оптимизации (20)–(22). Пользователь задает требования к точности программы путем указания значений параметров $\Delta P_{\tau}^*, \Delta P_{\tau \max \text{ доп}}$, которые могут быть любыми, и параметров $\Delta Y^*, \Delta Y_{\max \text{ доп}}$, которые должны удовлетворять следующим ограничениям:

$$\Delta Y^* \geq \frac{100}{199} \frac{\sum_{i=1}^m (P_{i \max} - P_{i \min})}{\sum_{i=1}^m P_{i \max}}; \quad \Delta Y_{\max \text{ доп}} \geq \frac{\sum_{i=1}^m P_{i \max} - P_{i \min}}{199}.$$

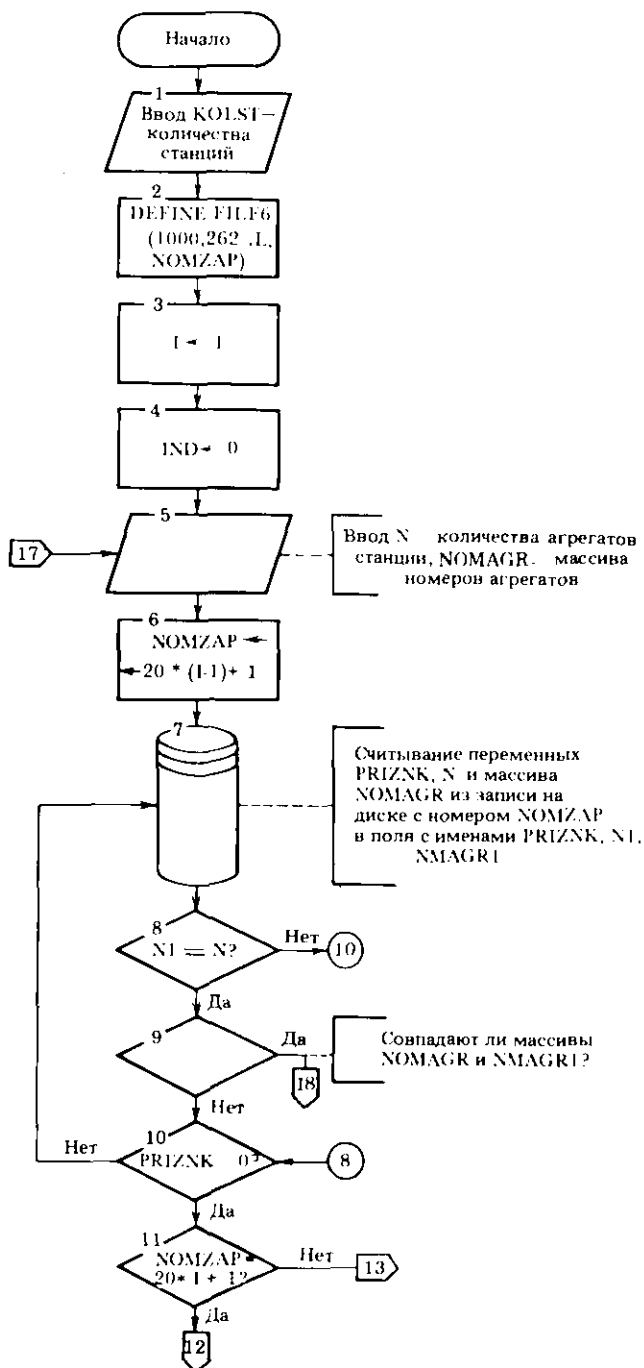


Рис. 34 (начало).

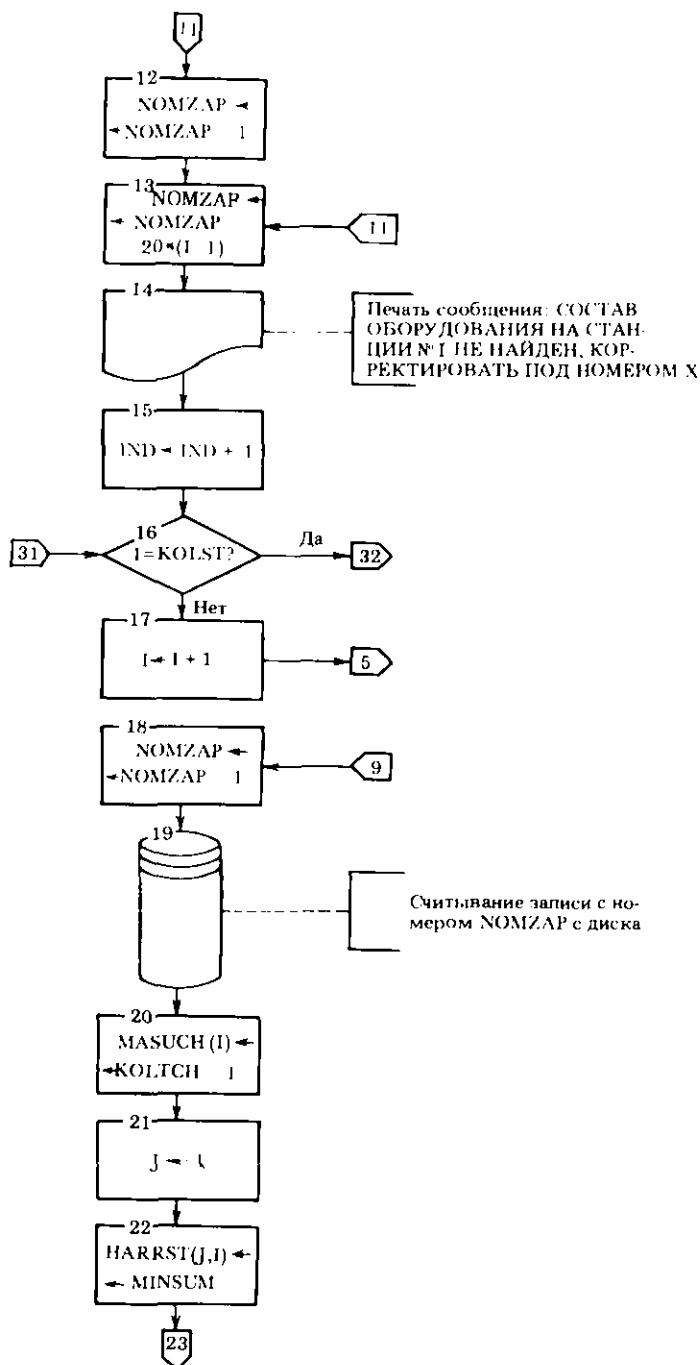


Рис. 34 (продолжение).

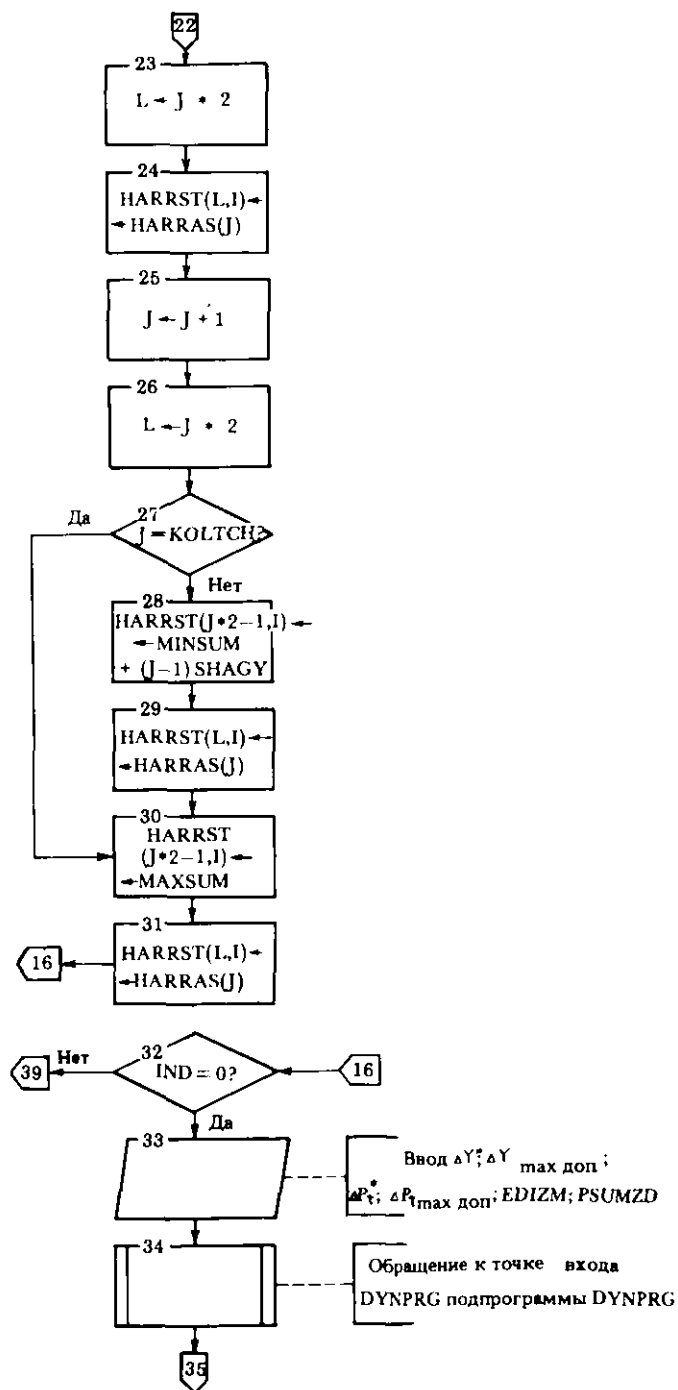


Рис. 34 (продолжение).

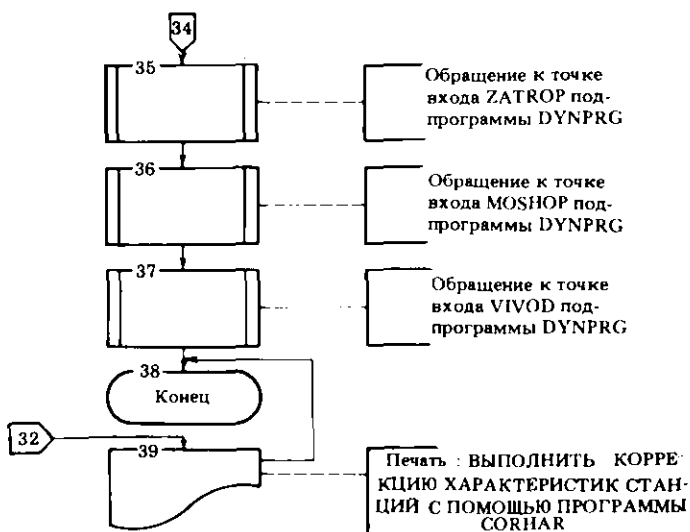


Рис. 34 (окончание)

При этом величины ΔY , ΔP_{τ} определяются как

$$\Delta Y = \min \left(\Delta Y^* \frac{\sum_{i=1}^m P_{i \max}}{100}, \Delta Y_{\max \text{ доп}} \right);$$

$$\Delta P_{\tau} = \min \left(\Delta P_{\tau}^* \frac{\max_{i=1, 2, \dots, m} P_{i \max}}{100}, \Delta P_{\tau \max \text{ доп}} \right).$$

Контрольный пример

Для иллюстрации использования программы RASNAGST при решении задачи (20)–(22) межстанционной оптимизации, а также форм входной и выходной информации, предусмотренных этой программой, в работе приведено решение контрольного примера. Образцы кодирования данных контрольного примера приведены в графах "Содержимое в контрольном примере" таблиц, представляющих формат карт управления и данных программы RASNAGST. Полностью исходные данные контрольного примера даны в параграфе "Ввод".

Ввод

Карты управления. Набор карт управления включает карту количества станций ККС, карту единицы измерения расходов топлива КЕИ, карту требований к точности решения задач КТТ и карты количеств включенных в работу агрегатов ККА1, ККА2, ..., ККА m для каждой станции энергосистемы. Форматы карт показаны соответственно в табл. 20, 21, 28, 29.

Карты данных. Набор карт данных включает карту заданной мощности потребления КЗМП (табл. 32) и карты номеров агрегатов на станциях КНА1, КНА2, ..., КНА m , формат которых показан в табл. 23.

Колонки перфокарты	Характеристика содержимого	Содержимое в контрольном примере
1–12	Заданная мощность потребления в системе	2.500E03
13–78	Пробелы	
79–80	Порядковый номер перфокарты в колоде	15

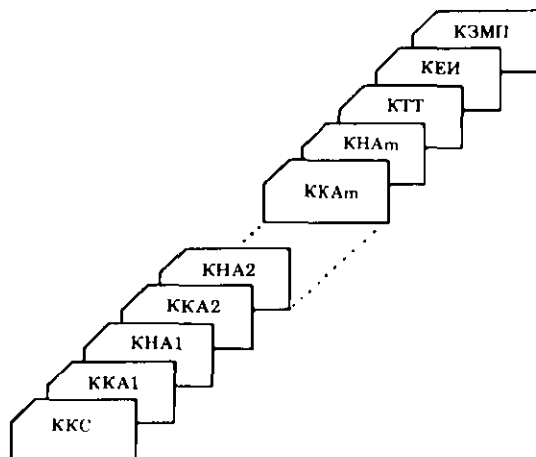


Рис. 35.

Колода перфокарт. Изображение колоды перфокарт управления и данных представлено на рис. 35. Распечатка вводимых карт для контрольного примера приведена на рис. 36.

Вывод

На печать выводятся следующие результаты решения задачи:

1) расход топлива в системе при оптимальном распределении активных мощностей между электростанциями;

2) оптимальные значения активных мощностей каждой станции.

Печать результатов контрольного примера приведена на рис. 37.

Инструкция по применению программы

Если программа RASNAGST каталогизирована в личную библиотеку абсолютных модулей, то задание на ее исполнение может иметь, например, следующий вид (соответствует примеру на исполнение программы CORHAR, приведенному в лабораторной работе 4):

```

// JOB РАСЧЕТКА
// ASSEN SYS004,X'00C'
// ASSEN SYS005,X'00F'
// EXEC CDDP

// UCP TD,FF,A=(00,82),B=(120),Q=(177,4)
// M1
// END

```

РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ

```

CARD TO PRINTER / PUNCH UTILITY
INPUT RECORD LENGTH 0030
OUTPUT RECORD LENGTH 0120
INPUT BLOCK LENGTH 0030
OUTPUT BLOCK LENGTH 00120
INPUT OPTION CARD BCD
OUTPUT OPTION PRINT CHARACTER
2 INPUT,2 OUTPUT AREAS ASSIGNED
RECORD FORMAT FIXED
TYPE DATA DISPLAY
STARTING SEQUENCE COLUMN 77
SEQUENCE LENGTH 04
STARTING RECORD NUMBER 00000001

```

РАСПЕЧАТКА КАРТ УПРАВЛЕНИЯ И ДАННЫХ

BL	SZ	BL NO	RC NO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	5																
2	2	1	3	4	5	2												
3	3	12																
4	4	6	5	1	8	2	10	12	3	11	4	9	7					
5	5	10																
6	6	6	5	8	1	9	7	3	2	4	10							
7	7	9																
8	8	1	2	3	4	5	7	6	9	8								
9	9	9																
10	10	8	2	9	1	6	3	4	5	7								
11	11	12																
12	12	9	2	8	3	7	4	6	5	10	12	11	1					
13	13		1,000	25,000	1,000	15,000												
14	14	14	ТОМЫ	УСЛ	ТОПЛИВА													
15	15	15	2,500E03															

END OF DATA

ЕОЗ РАСЧЕТКА РАСПЕЧАТКА ВЫПОЛНЕНА

Рис. 36.

12.37.12, DURATION 00.00.12

11.39.20

```
// JOB СТАНЦИ 'ОПТИМІЗАЦІЯ НАГРУЗОК СТАНЦИ'
// DLBL IJYS03,'FILE RASHARST'
// EXTENT SYS03,111111',,11,188
// ASSIGN SYS03,X'191'
// EXEC RASHARST
```

РАСХОД ТОПЛИВА= 465.071 ТОНН УСА ТОПЛИВА

I	I	I	I	I	I	I
I	НОМЕР	I	I	I	I	I
I	I	I	3	I	5	I
I	СТАНЦИ	I	I	I	I	I
I	I	I	I	I	I	I
I	I	I	I	I	I	I
I	I	I	I	I	I	I
I	НАГРУЗКА	I	I	I	I	I
I	I	I	I	I	I	I
I	I	591	I	416	I	481
I	(МВТ)	I	I	I	I	I
I	I	I	I	I	I	I

ЕОJ СТАНЦИ РЕШЕННЕ ЗАДАЧ ПО ВАРІАНТУ № ВПОЛНЕНО
Рис. 37.

11.40.20, DURATION 00.00.54

```
// JOB СТАНЦИИ'ОПТИМИЗАЦИЯ НАГРУЗОК СТАНЦИЙ'
  ASSIGN SYSCLB,X'191'
// DLBL IJSYSØ3,'ФАЙЛ РАСХОДНЫХ ХАРАКТЕРИСТИК
  СТАНЦИЙ'
//ASSGN SYSØØ3, X'191'
//EXEC RASNAGST
  (Файл карт управления и данных)
/*
  ASSIGN SYSCLB, UA
  /&-----РЕШЕНИЕ ЗАДАЧИ ПО ВАРИАНТУ
  N ВЫПОЛНЕНО
```

П р и м е ч а н и е. в позицию 42 карты /& необходимо занести номер указанного преподавателем варианта задания на выполнение лабораторной работы.

Сообщения об ошибках и время решения

Кроме стандартных сообщений операционной системы об ошибках, выдаются программные сообщения вида СОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ 1 НЕ НАЙДЕН, КОРРЕКТИРОВАТЬ ПОД НОМЕРОМ X для всех станций, для которых в файле не обнаружены расходные характеристики, соответствующие заданным составам включенных в работу агрегатов, а также итоговое сообщение ВЫПОЛНИТЬ КОРРЕКЦИЮ ХАРАКТЕРИСТИК СТАНЦИЙ С ПОМОЩЬЮ ПРОГРАММЫ CORNAR в случае появления хотя бы одного из сообщений описанного вида.

Время решения контрольного примера составляет 54 с.

Подпрограммы

Используемая в программе подпрограмма DYNPRG описана в лабораторной работе 1.

3.ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

При подготовке к выполнению лабораторной работы необходимо:

1) убедиться (см. лабораторную работу 2) в достаточном овладении материалом, относящимся к этапам обработки проблемных программ операционной системой ДОС/ЕС и к управляющим операторам программ УПРАВЛЕНИЕ ЗАДАНИЯМИ, РЕДАКТОР, БИБЛИОТЕКАРЬ;

2) убедиться (см. лабораторную работу 3), что усвоены вопросы использования управляющих операторов DLBL и EXTENT программы УПРАВЛЕНИЕ ЗАДАНИЯМИ, необходимых при работе с файлами на магнитных дисках;

3) изучить раздел 1 данной лабораторной работы, обратив особое внимание на зависимость текста задания, включающего этап редактирования, от характера ссылок между модулями, входящими в состав формируемой программы, а также на состав информации, необходимой пользователю при составлении заданий, использующих библиотеки;

4) изучить раздел 2 данной лабораторной работы, обратив особое внимание на вопросы подготовки колоды перфокарт для работы программы RASNAGST;

5) получить у преподавателя один из вариантов задания на выполнение данной лабораторной работы;

6) подготовить необходимую информацию на бланках по предложенному варианту задания для сдачи на перфорацию.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать: следующие разделы:

1) цель работы;

2) формулировку задачи оптимального распределения активных мощностей между электростанциями при заданных составах включенных в работу агрегатов;

3) форматы управляющих операторов DLBL и EXTENT программы УПРАВЛЕНИЕ ЗАДАНИЯМИ;

4) схему программы RASNAGST;

5) задание на трансляцию из личной библиотеки исходных модулей, редактирование и каталогизацию в личную библиотеку абсолютных модулей программы решения одной из подзадач задачи оптимизации режима энергосистемы по активной мощности (по соответствующему варианту задания на выполнение лабораторной работы);

6) задание операционной системе на исполнение программы RASNAGST, каталогизированной в личную библиотеку абсолютных модулей, для соответствующего варианта исходных данных;

7) сообщения операционной системы о результатах обработки задания на исполнение программы RASNAGST.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие преимущества обеспечивает использование модульного принципа при разработке комплекса программ для решения общей задачи?

2. В чем состоит характерная особенность структуры программ при использовании языка программирования ФОРТРАН IV?

3. Какую обязательную функцию осуществляет основная программа в случае многомодульной программы на ФОРТРАНе IV?

4. Для чего используются библиотеки исходных модулей при разработке комплекса программ?

5. Какие этапы обработки исходных модулей должны быть реализованы для получения готовых к исполнению программ решения подзадач?

6. Почему и каким образом должен учитываться характер ссылок между модулями внутри формируемой программы при составлении заданий операционной системе, включающих этап редактирования?

7. Какой управляющий оператор РЕДАКТОРА должен использоваться в случаях, когда автоматическое подключение нужного модуля в текст редактируемой программы посредством стандартного режима AUTOLINK невозможно?

8. Для чего необходима пользователю информация о регистрационных номерах пакетов дисков, на которых записаны личные библиотеки?

9. Как учитывается значение параметра "идентификатор файла", относящегося к файлу личной библиотеки, при составлении задания, включающего работу с этой библиотекой?

10. Для чего необходима пользователю информация о порядковом номере подбиблиотеки ФОРТРАНа IV в библиотеке исходных модулей?

11. Почему при трансляции исходных модулей из библиотеки необходимо последовательно назначить одно и то же физическое устройство ввода-вывода с магнитной лентой или магнитными дисками сначала логическому устройству SYSPCH, а затем SYSIPT?

12. Как обеспечивается в задании перемотка магнитной ленты к началу файла перед ее использованием в качестве логического устройства SYSIPT?

13. Какие функции при трансляции исходного модуля из библиотеки выполняет программа БИБЛИОТЕКАРя SSERV?

14. Куда помещается объектный модуль, формируемый в результате трансляции исходного модуля?

15. Какая системная программа осуществляет каталогизацию проблемной программы в личную библиотеку абсолютных модулей?

16. Каков состав условно-постоянной информации для подзадачи оптимального распределения активных мощностей между электростанциями в рассматриваемом комплексе программ?

17. Назовите целевую функцию и ограничения в формулировке задачи оптимального распределения мощностей между электростанциями.

18. Чем определяется точность решения задачи оптимального распределения мощностей между электростанциями методом динамического программирования?

19. Какие сообщения об ошибках и в каких случаях выдает программа RASNAGST?

20. С какой целью стандартная методика описания программ в литературе включает необходимость подробного описания обращений к используемым в программе подпрограммам?

6. ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Исходные данные для программы RASNAGST

Вариант 1

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ГРЭС-7	1	3	5	7		
ТЭЦ-4	6	4	3	8	2	
ТЭЦ-3	2	4	7	3	5	8
ГРЭС-5	5	4	7			

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 1\%$; $\Delta Y_{\text{max доп}} = 12 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ТЭЦ-2	2	1	5	4	8	
ГРЭС-1	6	3	5	1		
ТЭЦ-1	8	3	1	5	6	4
ГРЭС-2	2	7	4	6		

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,9\%$; $\Delta Y_{\max \text{ доп}} = 16 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ГРЭС-5	4	3	6			
ТЭЦ-2	6	2	5	7	1	
ГРЭС-2	5	3	7	4		
ТЭЦ-4	7	2	6	1	8	9
ГРЭС-3	1	5	7	2		

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\max \text{ доп}} = 23 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ГРЭС-5	6	4	7	5		
ТЭЦ-8	7	2	5	3	9	
ГРЭС-3	3	11	9			
ТЭЦ-5	2	5	11	7	9	10

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,9\%$; $\Delta Y_{\max \text{ доп}} = 20 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ГРЭС-8	5	9	7	4	3	
ТЭЦ-5	6	4	7	9	8	1
ТЭЦ-4	1	9	8	2	4	6

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\max \text{ доп}} = 14 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов					
ТЭЦ-1	1	5	2	3		
ТЭЦ-7	7	2	8	4	10	
ГРЭС-2	5	4	1			
ГРЭС-9	2	8	4	6		
ГРЭС-5	7	1	9			

У к а з а н и е. Задание выполнить при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\max \text{ доп}} = 22 \text{ МВт}$.

Наименование станции	Эксплуатационные номера включенных в работу агрегатов						
ТЭЦ-1	2	4	10	11	7	3	5
ГРЭС-2	5	2	7	9	10		
ГРЭС-3	11	7	3	4	5		

У к а з а н и е. Задание выполнять при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\text{max доп}} = 12 \text{ МВт}$.

Вариант 8

Наименование станции	Эксплуатационные номера включенных в работу агрегатов						
ТЭЦ-2	2	1	7	6	12	8	
ГРЭС-7	1	5	3	4			
ГРЭС-4	1	7	2	4			
ТЭЦ-8	1	7	4	9	10		

У к а з а н и е. Задание выполнять при условиях: $\Delta Y^* = 0,8\%$; $\Delta Y_{\text{max доп}} = 18 \text{ МВт}$.

Каталогизируемые программы

Имена программ, каталогизируемых в личные БАМ, в вариантах заданий следующие:

RASNAGA1 (к варианту 1);
 POSHARST (к варианту 2);
 CORHAR (к варианту 3);
 RASNAGST (к варианту 4);
 RASNAGA1 (к варианту 5);
 POSHARST (к варианту 6);
 CORHAR (к варианту 7);
 RASNAGST (к варианту 8).

Идентификаторы файлов личных библиотек

Личные библиотеки исходных и абсолютных модулей имеют следующие идентификаторы в вариантах заданий:

SYSSLB - 'БИБЛИОТЕКА_ИСХОДНЫХ_МОДУЛЕЙ'; SYSCLB -
 'БИБЛИОТЕКА_АБСОЛЮТНЫХ_МОДУЛЕЙ' (к варианту 1);
 SYSSLB - 'IJSYSSL', SYSCLB - 'БИБЛИОТЕКА_АБСОЛЮТНЫХ_МОДУЛЕЙ' (к варианту 2);
 SYSSLB - 'ИСХОДНЫЕ_МОДУЛИ', SYSCLB - 'IJSYSSL' (к варианту 3);
 SYSSLB - 'КНИГИ', SYSCLB - 'ФАЗЫ' (к варианту 4);
 SYSSLB - 'IJSYSSL', SYSCLB - 'IJSYSSL' (к варианту 5);
 SYSSLB - 'КНИГИ', SYSCLB - 'IJSYSSL' (к варианту 6);
 SYSSLB - 'IJSYSSL', SYSCLB - 'ФАЗЫ' (к варианту 7);
 SYSSLB - 'IJSYSSL', SYSCLB - 'IJSYSSL' (к варианту 8);

Примечания 1. По каждому варианту составить задание для ДОС/ЕС на трансляцию из личной библиотеки исходных модулей, редактирование и каталогизацию в личную библиотеку абсолютных модулей программы, указанной для соответствующего варианта в параграфе "Каталогизируемые программы". Модульная структура программы дана в разделе 1. Регистрационный номер пакета магнитных дисков, содержащего файлы личных библиотек, - ABCDEF. Номер подбиблиотеки ФОРТРАНа IV в библиотеке исходных модулей - 3.

2. По каждому варианту составить задание на исполнение программы RASNAGST, каталогизированной в личную библиотеку абсолютных модулей, учитывая структуру файла расходных характеристик, подготовленного при выполнении лабораторных работ 3, 4.

ОРГАНИЗАЦИЯ ЭКСПЛУАТАЦИИ, ПОПОЛНЕНИЯ И ОБНОВЛЕНИЯ КОМПЛЕКСА ПРОГРАММ НА ПРИМЕРЕ ЗАДАЧИ ОПТИМИЗАЦИИ РЕЖИМА ЭНЕРГОСИСТЕМЫ ПО АКТИВНОЙ МОЩНОСТИ

Ц е л ь р а б о т ы: ознакомиться на примере задачи оптимизации режима энергосистемы по активной мощности с приемами выбора последовательности работы программ комплекса для решения подзадач и режимов работы этих программ в зависимости от ситуаций, возникающих в процессе эксплуатации энергосистемы. Кроме того, работа позволит ознакомиться с предоставляемыми операционной системой ДОС/ЕС простейшими возможностями пополнения комплекса программ новыми программными модулями с целью его совершенствования и замены устаревшего комплекса новым.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

В процессе эксплуатации задачи могут возникать различные ситуации, требующие от пользователя выбора наиболее рационального способа автоматизированного решения задачи. На организацию процесса ее решения могут влиять такие факторы, как располагаемое время для решения задачи по условиям текущей оперативной обстановки в энергосистеме; степень информированности пользователя о текущем состоянии файлов условно-постоянной информации (УПИ) по данной задаче; наличие или отсутствие запросов со стороны административных и производственных подразделений энергосистемы на получение той или иной информации; длительность качественных изменений режима энергосистемы в связи с выводом агрегатов в резерв, текущий или капитальный ремонт; ввод в эксплуатацию новых агрегатов или новых электростанций и т.д. В зависимости от этих факторов изменяется рациональный порядок использования программ решения различных подзадач, входящих в состав комплекса программ, образующего математическое обеспечение данной задачи, в частности, могут изменяться подмножество необходимых программ в создавшейся ситуации, последовательность их исполнения, настройка на нужный режим и т.д.

Для обеспечения необходимой оперативности и надежности решения задачи пользователь должен, исходя из конкретных особенностей задачи, состава комплекса программ и их возможностей, заранее определить четкий порядок использования комплекса во всех типичных для нормальной эксплуатации ситуациях. Пусть, например, в связи с предстоящим наступлением ремонтного периода ожидается существенное изменение составов агрегатов, включаемых в работу на электростанциях для покрытия суточных графиков нагрузки. В этом случае для обеспечения успешной эксплуатации на протяжении указанного периода комплекса программ оптимизации режима энергосистемы по активной мощности, рассматриваемого в настоящем лабораторном практикуме, целесообразно в один из периодов времени, свободных от оперативного решения задачи оптимизации, выполнить программу CORHAR (см. лабораторную работу 4) в режиме ОТПЕЧАТАТЬ СОСТАВЫ для всех тех электростан-

ций, где возможно существенное изменение составов включаемого в работу оборудования. Проанализировав полученную информацию, пользователь может определить, для каких электростанций и ожидаемых составов агрегатов на них расходные характеристики в файле отсутствуют. После этого должна быть подготовлена соответствующая исходная информация для исполнения программы CORHAR в режиме СКОРРЕКТИРОВАТЬ. Выполнение на ЭВМ программы CORHAR в этом режиме позволит построить отсутствующие расходные характеристики и записать их в файл УПИ для данной задачи.

В связи с постоянным ростом и совершенствованием энергосистем программные комплексы непрерывно развиваются как по линии использования уточненных и более адекватных реальным технологическим процессам математических моделей при решении производственных задач (это повышает эффективность использования народнохозяйственных ресурсов), так и по линии включения в состав комплексов программ все более совершенных сервисных функций, обеспечивающих своевременность и полноту удовлетворения предъявляемых пользователями запросов по данной задаче.

Так, рассматриваемый в данной работе комплекс программ оптимизации режима энергосистемы по активной мощности может быть дополнен программой, осуществляющей оптимальный выбор состава включенных в работу агрегатов на электростанциях при заданных суммарной нагрузке энергосистемы и составе агрегатов, находящихся в горячем резерве. Для улучшения сервисных функций комплекса можно, например, дополнить его программой, выдающей в отличие от программы CORHAR справки о запрашиваемых расходных характеристиках станций независимо от того, имеются ли они в файле расходных характеристик для подзадачи межстанционной оптимизации.

Операционная система ДОС/ЕС создает возможности для постепенного совершенствования или пополнения используемых комплексов программ при решении технологических задач. Для выполнения этой функции пользователю достаточно после трансляции и редактирования вновь разработанных программ каталогизировать их в системную либо личную библиотеку абсолютных модулей (см. лабораторные работы 2, 5).

В процессе эксплуатации может возникнуть и необходимость в полной замене используемого комплекса программ новым, более совершенным. Так, алгоритм оптимизации режима энергосистемы по активной мощности может быть разработан и программно реализован таким образом, чтобы учитывалась зависимость потерь активной мощности в сети от мощностей станций. Другой пример возможного совершенствования алгоритма — совмещение оптимизации активных мощностей в энергосистеме с оптимизацией распределения тепловых нагрузок между агрегатами ТЭЦ. Очевидно, в таких случаях придется полностью перестроить весь рассматриваемый комплекс программ для решения данной задачи.

Если эксплуатировавшийся комплекс программ хранился в личной библиотеке абсолютных модулей, то, возможно, для его замены или обновления потребуется создать новую библиотеку. Функцию создания системных и личных библиотек по соответствующим заданиям пользователя выполняет одна из системных программ БИБЛИОТЕКАРЯ (см. лабораторную работу 2), каталогизированная в системной библиотеке абсолютных модулей под именем CORGZ.

Создание личной библиотеки исходных, объектных или абсолютных модулей заключается в осуществлении необходимой разметки выбранного пользователем участка пакета магнитных дисков, на котором предполагается хранить создаваемую библиотеку, и в формировании на определенной части этого участка стандартного набора записей, образующего оглавление библиотеки. Оглавление предназначено для хранения информации о каждой из каталогизируемых в библиотеку программ (эта информация необходима для вызова соответствующей программы по заданию пользователя).

После исполнения программой CORGZ функции создания библиотеки последняя становится доступной для каталогизации модулей соответствующего типа.

В задании пользователя на создание библиотеки, как и в любом задании на исполнение программы, обрабатывающей файлы на магнитных дисках, должно присутствовать описание файла с помощью управляющих операторов DLBL и EXTENT программы УПРАВЛЕНИЕ ЗАДАНИЯМИ (см. лабораторную работу 3), сообщающих операционной системе информацию, необходимую для формирования меток файла, который отводится под создаваемую библиотеку.

Для файлов библиотек, как и для файлов проблемных программ ФОРТ-РАНа IV (см. лабораторную работу 3), разрешается отводить только один последовательный участок пакета магнитных дисков. Для задания в операторе EXTENT значения параметра "количество дорожек в участке" пользователь должен определить размеры файла исходя из следующего: является ли создаваемая личная библиотека библиотекой исходных, объектных или абсолютных модулей; сколько и каких конкретных программ предполагается поместить в библиотеку.

Общая формула для определения количества дорожек файла, отводимого под библиотеку, имеет вид

$$K = L_0 + L_1, \quad (23)$$

где L_0 — размер оглавления; L_1 — размер собственно библиотеки.

Размер оглавления для каждой из библиотек зависит только от максимального количества N независимо каталогизируемых программ, которые предполагается в дальнейшем поместить в библиотеку:

библиотека исходных и объектных модулей

$$L_0 = \begin{cases} \frac{N+6}{160}, & \text{если } (N+6)/160 - \text{целое число,} \\ E\left(\frac{N+6}{160}\right) + 1, & \text{если } (N+6)/160 - \text{дробное;} \end{cases} \quad (24)$$

библиотека абсолютных модулей

$$L_0 = \begin{cases} \frac{N+1}{144} + 10, & \text{если } (N+1)/144 - \text{целое число.} \\ E\left(\frac{N+1}{144}\right) + 11, & \text{если } (N+1)/144 - \text{дробное.} \end{cases} \quad (25)$$

Размер собственно библиотек зависит не только от количества программ, но и от объема каждой из них. Поскольку при создании библиотек точная структура подлежащих каталогизации программ обычно еще не определена, можно ориентировочно рекомендовать

$$L_i \approx n_0 N / (10-20),$$

где n_0 -- среднее количество операторов исходных модулей на одну программу.

В качестве значения параметра "адрес участка в пакете" в операторе EXTENT для случая библиотек исходных и объектных модулей можно задавать любую дорожку с единственным ограничением: файл с библиотекой не должен выходить за пределы 199-го цилиндра пакета. Для библиотеки абсолютных модулей дополнительное ограничение состоит в том, что номер дорожки должен быть кратным 10.

В качестве значения параметра "имя файла" в операторе DLBL должны быть заданы соответственно идентификаторы IJYSSL для библиотеки исходных модулей, IJSYRL для библиотеки объектных модулей и IJSYSPC для библиотеки абсолютных модулей. Указанное в лабораторной работе 3 для библиотек однозначное соответствие имен системных файлов и логических устройств ввода-вывода не имеет места, пока библиотеки не созданы. При создании библиотек в ДОС/ЕС-2.1 предусмотрено следующее соответствие логических устройств файлам создаваемых личных библиотек: для библиотеки исходных модулей -- SYSSLB; для библиотеки объектных модулей -- SYSRLB; для библиотеки абсолютных модулей -- SYSØØ3.

Операторы DLBL и EXTENT, как и обычно, используются системой ввода-вывода. Для работы программы CORGZ, создающей библиотеку, пользователь дополнительно должен сообщить в задании операционной системе информацию о структуре создаваемой библиотеки. Для этой цели служит управляющий оператор БИБЛИОТЕКАРЯ NEWVOL. Его формат:

NEWVOL идентификатор библиотеки = размер библиотеки (размер оглавления)

Параметр "идентификатор библиотеки", определяющий тип создаваемой библиотеки, должен принимать следующие значения: для библиотеки исходных модулей -- SL; для библиотеки объектных модулей -- RL; для библиотеки абсолютных модулей -- CL.

Параметру "размер библиотеки" в качестве значения задается количество цилиндров пакета магнитных дисков, отводимое под файл библиотеки. Оно вычисляется как

$$Z = \begin{cases} K/10, & \text{если } K/10 - \text{целое число,} \\ E(K/10) + 1, & \text{если } K/10 - \text{дробное,} \end{cases}$$

где K определяется по (23).

Параметру "размер оглавления" в качестве значения задается число L_0 дорожек, предназначенных для записи оглавления библиотеки, определяемой по (24), (25).

В тексте задания оператор NEWVOL должен следовать за управляющим оператором программы УПРАВЛЕНИЕ ЗАДАНИЯМИ // EXEC CORGZ, по которому осуществляется вызов для исполнения программы CORGZ. В резуль-

тате исполнения этой программы, помимо созданной библиотеки, пользователь получает распечатку ее оглавления на SYSLST.

Пример 1. Составить задание для ДОС/ЕС на создание библиотеки исходных модулей, предназначенной для хранения 10 модулей ФОРТРАНа IV, содержащих приблизительно по 300 операторов каждый.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB СОЗДБИБ1
// ASSIGN SYSSLB,X'191'
// DLBL IJSYSSL,УЧЕТ_ЗАПАСОВ_ТОПЛИВА',1830
// EXTENT SYSSLB, ЗАПАСЫ,,,5,200
// EXEC CORGZ
NEWVOL SL=20(1)
/&
```

Пример 2. Составить задание для ДОС/ЕС на создание библиотеки объектных модулей, предназначенной для хранения пяти транслированных модулей, имеющих на исходном языке ФОРТРАН IV примерно по 400 операторов каждый.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB СОЗДБИБ2
// ASSIGN SYSRLB,X'191'
// DLBL IJSYSSL,'СТАТИСТИКА_ОТКАЗОВ', 366
// EXTENT SYSRLB, ОТКАЗЫ,,,17,100
// EXEC CORGZ
NEWVOL RL=10(1)
/&
```

Пример 3. Составить задание для ДОС/ЕС на создание библиотеки абсолютных модулей, предназначенной для хранения 20 фаз, каждой из которых на исходном языке соответствует примерно по 1000 операторов.

Задание операционной системе, составленное в соответствии с условиями примера, имеет вид:

```
// JOB СОЗДБИБ3
// ASSIGN SYS003,X'191'
// DLBL IJSYSPC, 'ПЕРЕХОДНЫЕ ПРОЦЕССЫ', 731
// EXTENT SYS003,1000000,,,30,1000
// EXEC CORGZ
NEWVOL CL=100(11)
/&
```

2. КОМПЛЕКС ПРОГРАММ ДЛЯ ЕС ЭВМ ЗАДАЧИ ОПТИМИЗАЦИИ РЕЖИМА ЭНЕРГОСИСТЕМЫ ПО АКТИВНОЙ МОЩНОСТИ

В данном разделе приводятся сведения о программах для решения всех подзадач оптимизации режима энергосистемы по активной мощности, предусмотренных изучаемым в данном лабораторном практикуме программным комплексом, в объеме, достаточном для создания соответствующих библиотек исходных, объектных и абсолютных модулей, а также для выбора рационального порядка работы программ комплекса в зависимости от возникающих в производственном процессе ситуаций.

В подзадаче оптимального распределения активных мощностей между агрегатами электростанций основная программа ФОРТРАНа IV — RASNAGA1, количество операторов 22. Используемая подпрограмма — DYNPRG, количество операторов 230. Режим работы программы единственный, производится распределение заданной суммарной нагрузки между агрегатами электростанций.

В подзадаче организации файла расходных характеристик электростанций основная программа ФОРТРАНа IV--POSHARST, количество операторов 3. Используемые подпрограммы: HARST, количество операторов 80; PESNAT, количество операторов 80; DYNPRG, количество операторов 230.

Режимы работы программы:

1) ТОЛЬКО ОТПЕЧАТАТЬ ХАРАКТЕРИСТИКИ.

Для всех станций энергосистемы производится построение и вывод на печать по одной расходной характеристике, соответствующей заданному составу включенных в работу агрегатов;

2) ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ.

Действия те же, что и в предыдущем режиме, но характеристики не выводятся на печать, а записываются в файл расходных характеристик на пакете магнитных дисков;

3) ОТПЕЧАТАТЬ И ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ.

Совмещены функции режимов ТОЛЬКО ОТПЕЧАТАТЬ ХАРАКТЕРИСТИКИ и ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ.

В подзадаче коррекции файла расходных характеристик основная программа ФОРТРАНа IV--CORHAR, количество операторов 100. Используемые подпрограммы: HARST, количество операторов 80; PESNAT, количество операторов 80; DYNPRG, количество операторов 230.

Режимы работы программы следующие:

1) ИСКАТЬ И СООБЩИТЬ. Для заданного подмножества электростанций энергосистемы и указанного на каждой из них состава включенных в работу агрегатов выдается сообщение о том, имеется ли соответствующая расходная характеристика в файле;

2) ОТПЕЧАТАТЬ СОСТАВЫ. Для заданного подмножества электростанций энергосистемы производится печать всех составов включенных в работу агрегатов, которые соответствуют расходным характеристикам станций, имеющимся в файле;

3) ОТПЕЧАТАТЬ ИЛИ СООБЩИТЬ. Для заданного подмножества электростанций энергосистемы и указанного на каждой из них состава включенных в работу агрегатов производится печать расходных характеристик, если они имеются в файле, или печать сообщения об отсутствии характеристик в противном случае;

4) СКОРРЕКТИРОВАТЬ. Для заданного подмножества электростанций энергосистемы производится запись в файл расходных характеристик, соответствующих указанным составам включенных в работу агрегатов. При этом возможны две модификации режима (в зависимости от того, используется ли для подпрограммы HARST режим ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ либо режим ОТПЕЧАТАТЬ И ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ): а) записываемые в файл характеристики не выводятся на печать; б) производится печать записываемых в файл характеристик.

В подзадаче оптимального распределения активных мощностей между электростанциями энергосистемы основная программа ФОРТРАНа IV--RASNAGST, количество операторов 65. Используемая подпрограмма -- DYNPRG, количество операторов 230. Режим работы единственный: если для всех электростанций энергосистемы расходные характеристики для заданных

составов включенных в работу агрегатов имеются в файле, производится распределение суммарной мощности потребления в энергосистеме между электростанциями; в противном случае соответствующее число раз выдается сообщение об ошибке пользователя, подготовившего задание на исполнение программы.

3. ЗАДАНИЕ НА ПРЕДВАРИТЕЛЬНУЮ ПОДГОТОВКУ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ

С учетом указанной в начале данной лабораторной работы цели при подготовке к выполнению ее необходимо:

1) изучить раздел 1 данной лабораторной работы, обратив особое внимание на сущность проблем, возникающих перед пользователем задачи в процессе текущей эксплуатации комплекса программ, предназначенных для ее решения; на методы пополнения и полного обновления комплекса программ; на методику составления заданий на создание личных библиотек исходных, объектных и абсолютных модулей;

2) изучить раздел 2 данной лабораторной работы;

3) используя при необходимости лабораторную работу 5, убедиться в понимании причин и характера различий в составе комплексов исходных, объектных и абсолютных модулей для решения одной и той же конкретной производственной задачи;

4) получить у преподавателя один из вариантов задания на выполнение данной лабораторной работы.

4. СОДЕРЖАНИЕ ОТЧЕТА

Отчет должен включать следующие разделы:

1) цель работы;

2) характеристику комплекса программ для решения задачи оптимизации режима энергосистемы по активной мощности;

3) формулировку варианта задания на выполнение лабораторной работы и четко оформленное решение по этому заданию.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Какие факторы, возникающие в процессе эксплуатации задачи оптимизации режима энергосистемы по активной мощности, влияют на выбор рациональной последовательности исполняемых программ решения подзадач и выбор их режимов?

2. Чем объясняется необходимость пополнения комплекса программ для решения производственной задачи новыми модулями?

3. Какие возможности операционной системы надо использовать, чтобы включить в комплекс новую программу?

4. Какие возможности операционной системы позволяют полностью обновить используемый комплекс программ для решения задачи?

5. Какая системная программа выполняет функцию создания личных библиотек?

6. Как определить размеры файлов для личных библиотек исходных, объектных и абсолютных модулей?

7. В чем заключаются особенности задания значений параметров для управляющих операторов программы УПРАВЛЕНИЕ ЗАДАНИЯМИ, служащих для описания файлов на магнитных дисках в случае файлов, отводимых под личные библиотеки?

8. Какой управляющий оператор БИБЛИОТЕКАРЯ используется в ДОС/ЕС-2.1 при создании личных библиотек?

9. Какое условие налагается на порядок размещения управляющих операторов в задании операционной системе на создание личной библиотеки?

10. В чем состоит различие составов личных библиотек исходных, объектных и абсолютных модулей для решения одной и той же задачи?

11. Какую информацию выдает операционная система пользователю при обработке заданий на создание библиотек?

6. ВАРИАНТЫ ЗАДАНИЙ НА ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Вариант 1

В энергосистеме предстоит ввод нового агрегата на одной из электростанций. Определить наиболее рациональный состав и последовательность работы подлежащих исполнению программ решения подзадач задачи оптимизации режима энергосистемы по активной мощности, используемые при исполнении программ режимы и состав исходной информации с целью заблаговременной подготовки к оперативной эксплуатации данной задачи в новых условиях.

Вариант 2

Данные те же, что в варианте 1, но в ситуации, когда в энергосистеме предстоит ввод в эксплуатацию новой электростанции.

Вариант 3

Данные те же, что в варианте 1, но в следующей ситуации: в условиях текущей эксплуатации необходимо получить от ЭВМ информацию об оптимальных мощностях станций при заданной мощности потребления в энергосистеме и заданных составах включенных в работу агрегатов на каждой станции, о расходных характеристиках станций для этих составов оборудования и об оптимальных нагрузках каждого агрегата, причем известно, что требуемые расходные характеристики имеются в файле.

Вариант 4

Данные те же, что в варианте 3, но не исключено отсутствие в файле некоторых расходных характеристик из числа требуемых.

Вариант 5

Данные те же, что в варианте 1, но в ситуации, когда до рассматриваемого момента комплекс программ оптимизации режима энергосистемы по активной мощности в данной системе не эксплуатировался.

Вариант 6

Составить задание для ДОС/ЕС-2.1 на создание личной библиотеки исходных модулей, предназначенной для хранения комплекса программ оптимизации режима энергосистемы по активной мощности.

Вариант 7

Данные те же, что в варианте 6, но для библиотеки объектных модулей.

Вариант 8

Данные те же, что в варианте 6, но для библиотеки абсолютных модулей.

ЛИТЕРАТУРА

- Автоматизация управления энергообъединениями/ В.В. Гончуков, В.М. Горнштейн, Л.А. Крумм и др.; Под ред. С.А.Совалова. - М.: Энергия, 1979. - 432 с.
- Айнберг В.Д., Геронимус Ю.В. Основы программирования для Единой Системы ЭВМ. - М.: Машиностроение, 1980. - 336 с.
- Арзамасцев Д.А., Бартоломей П.И., Холян А.М. АСУ и оптимизация режимов энергосистем. - М.: Высш. шк., 1983. - 208 с.
- Веников В.А., Журавлев В.Г., Филиппова Т.А. Оптимизация режимов электростанций и энергосистем. - М.: Энергоиздат, 1981. - 464 с.
- Дроздов Е.А., Комарницкий В.А., Пятибратов А.П. Электронные вычислительные машины Единой системы. - М.: Машиностроение. 1976. - 672 с.
- Операционная система ДОС/ЕС: Справочник/Ю.Ю.Битель, В.И.Воюш, Р.В.Горбунова и др. - М.: Статистика, 1978. - 270 с.
- Оптимизация режимов энергетических систем/А.В.Богословский, В.Г.Григоренко, Я.А.Калиновский; Под ред. В.М.Синькова. - Киев: Вища шк., 1976. - 308 с.
- Падалко Л.П., Пекелис Г.Б. Экономика энергосистем. - Минск: Высш. шк., 1976. - 484 с.
- Переход от ДОС ЕС к ОС ЕС/О.С.Иванико, Н.М.Иванютина, М.П.Котов и др. - М.: Статистика, 1980. - 231 с.
- Электрические системы: Автоматизированные системы управления режимами энергосистем/ В.А. Богданов, В.А. Веников, Я.Н. Лугинский, Г.А. Черня; Под ред. В.А.Веникова. - М.: Высш. шк., 1979. - 448 с.

ПРИЛОЖЕНИЕ

Программа RASNAGA 1

=====

```

DIMENSION EDIZM(4)
INTEGER N*4, A*2(20), OTVET*4(20), NOMAGR*2(20), TEXT*2(4) / 'AG', 'PE',
C 'GA', 'TA' /
REAL*4 IZLON(100, 20), SHAGY, MSHAGY, SHAGP, MSHAGP, HAR(200, 20)
READ(1, 1) N, EDIZM, (NOMAGR(I), A(I), I=1, N)
1 FORMAT(13/4A4/26(13))
DATA KOLUCH/100/, KOLZNY/200/
DO 2 I=1, N
K=2*(A(I)+1)
READ(1, 3) (IZLON(J, I), J=1, K)
3 FORMAT(3(E7, 0, E15, 3))
2 CONTINUE
READ(1, 4) SHAGY, MSHAGY, SHAGP, MSHAGP
4 FORMAT(4(E9, 2))
READ(1, 92) PSUMZD
92 FORMAT(E12, 3)
CALL DYNPRG(N, A, KOLUCH, IZLON, SHAGY, MSHAGY, SHAGP, MSHAGP, OTVET, KOLZNY, HAR)
CALL ZATROP(PSUMZD, IZLON, 80PT)
CALL MOSHOP(PSUMZD, IZLON, HAR, OTVET)
CALL VIVOD(EDIZM, TEXT, OTVET, NOMAGR)
STOP
END

```

Программа POSHARST

=====

```

CALL HARST(1)
STOP
END

```

Программа CORHAR

=====

```

INTEGER*2 KOLST, MASNST(50), NOMAGR(20), PRIZNK, NMAGR(20), NMCRHR(50)
C, UPRCAR(12), ETALON(48) / ИСКАТЬ И СООБЩИТЬ ОТПЕЧАТАТЬ ИЛИ СОО
СБЩИТЬ ОТПЕЧАТАТЬ СОСТАВЫ СКОРРЕКТИРОВАТЬ ' /, IND / ' 01' /
REAL HARRAS(50), MINSUM, MAXSUM
READ(1, 1) UPRCAR, KOLST, (MASNST(I), I=1, KOLST)
1 FORMAT(12A2/13/25(13))
DEFINE FILE 6(1000, 262, L, NOMZAP)
DO 2 I=1, 12
IF(UPRCAR(I).NE.ETALON(1))GO TO 3
2 CONTINUE
29 I=1
21 DO 4 J=1, 12
IF(UPRCAR(J).NE.ETALON(24+J))GO TO 5
4 CONTINUE
GO TO 6

```

```

5 READ(1,7)N,(NOMAGR(J),J=1,N)
7 FORMAT(I3/2#(I3))
6 NOMZAP=2#*(MASNST(I)-1)+1
17 READ(6,NOMZAP)PRIZNK,N1,(NMAGR1(J),J=1,N1)
DO 8 J=1,12
IF(UPRCAR(J).NE.ETALON(24+J))GO TO 9
8 CONTINUE
GO TO 1#
9 IF(N1.NE.N)GO TO 11
DO 12 J=2,N
K=J-1
DO 13 L=1,K
IF(NOMAGR(K+2-L).GT.NOMAGR(K+1-L))GO TO 12
NOMPR=NOMAGR(K+1-L)
NOMAGR(K+1-L)=NOMAGR(K+2-L)
13 NOMAGR(K+2-L)=NOMPR
12 CONTINUE
DO 14 J=1,N
IF(NOMAGR(J).NE.NMAGR1(J))GO TO 11
14 CONTINUE
NOMZAP=NOMZAP-1
GO TO 15
10 WRITE(3,16)MASNST(I),(NMAGR1(J),J=1,N1)
16 FORMAT(/1H ,T43,32НСОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N,12,1H ,T43,13
СНАГРЕГАТЫ NN: ,2#(I2,1H,))
IF(PRIZNK.EQ.1)GO TO 17
GO TO 26
11 IF(PRIZNK.EQ.1)GO TO 17
IF(NOMZAP.NE.2#*MASNST(I)+1)GO TO 18
NOMZAP=NOMZAP-1
18 NOMHAR=NOMZAP-2#*(MASNST(I)-1)
WRITE(3,19)MASNST(I),NOMHAR
19 FORMAT(/1H ,T23,32НСОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N,12,3H НЕ НАЙ-
САЕН,КОРРЕКТИРОВАТЬ ПОА НОМЕРОМ ,12)
26 IF(1.NE.KOLST)GO TO 2#
36 STOP
2# I=I+1
GO TO 21
15 DO 22 J=1,12
IF(UPRCAR(J).NE.ETALON(12+J))GO TO 23
22 CONTINUE
GO TO 24
23 NOMHAR=NOMZAP-20*(MASNST(I)-1)
WRITE(3,25)MASNST(I),NOMHAR
25 FORMAT(/1H ,T34,32НСОСТАВ ОБОРУДОВАНИЯ НА СТАНЦИИ N,12,2H НАААЕН
С ПОА НОМЕРОМ ,12)
GO TO 26
24 READ(6,NOMZAP)PRIZNK,N1,(NOMAGR(J),J=1,N),KOLTCH,MINSUM,MAXSUM,
CS HAGY,(HARRAS(J),J=1,KOLTCH)
NOMST=MASNST(I)
CALL PECHAT(HARRAS,KOLTCH,MINSUM,MAXSUM,SHAGY,NOMST,NOMAGR,N,EDIZM
C)
GO TO 26
3 DO 27 J=1,12
IF(UPRCAR(J).NE.ETALON(12+J))GO TO 28
27 CONTINUE
GO TO 24
28 DO 30 J=1,12
IF(UPRCAR(J).NE.ETALON(24+J))GO TO 31
3# CONTINUE
GO TO 29
31 DO 32 J=1,12
IF(UPRCAR(J).NE.ETALON(36+J))GO TO 33

```

```

32 CONTINUE
GO TO 34
33 WRITE(3,35)
35 FORMAT(1#(/),1H ,Т47,27НОШИБОЧНАЯ УПРАВЛЯЮЩАЯ КАРТА)
GO TO 36
34 READ(1,37)(NMCRHR(J),J=1,KOLST)
37 FORMAT(26(13))
I=1
38 NOMST=MASNST(I)
NOMHAR=NMCRHR(I)
CALL HARST1(IND,NOMST,NOMHAR)
NOMZAP=20*(MASNST(I)-1)+NOMHAR-1
READ(6,NOMZAP)PRIZNK,N,(NOMAGR(J),J=1,N),KOLTCH,MINSUM,MAXSUM,SHAGY,
CCHGY,(HARRAS(J),J=1,KOLTCH)
PRIZNK=1
NOMZAP=NOMZAP-1
WRITE(6,NOMZAP)PRIZNK,N,(NOMAGR(J),J=1,N),KOLTCH,MINSUM,MAXSUM,
CCHGY,(HARRAS(J),J=1,KOLTCH)
IF(I,EQ,KOLST)GO TO 36
I=I+1
GO TO 38
END

```

Программа RASNAGST

=====

```

INTEGER=2 MASNST(50),NOMAGR(20),PRIZNK,NMAGR1(20),OTVET=4(50),
CTEXT=2(4)/8H СТАЦИИ/,IND,MASUCH(50),KOLST=4
REAL HARRAS(50),HARRST(100,50),MINSUM,MAXSUM,SHAGY,MSHAGY,SHAGP,
CMSHAGP,HAR(200,50)
DIMENSION EDIZH(4)
DATA KOLUGH/100/,KOLZNY/200/
READ(1,1) KOLST,(MASNST(I),I=1,KOLST)
1 FORMAT(13/25(13))
DEFINE FILE 6(1000,262,L,NOMZAP)
I=1
IND=0
12 READ(1,2) N,(NOMAGR(J),J=1,N)
2 FORMAT(13/25(13))
NOMZAP=20*(MASNST(I)-1)+1
8 READ(6,NOMZAP) PRIZNK,N1,(NMAGR1(J),J=1,N1)
IF(N1.NE.N)GO TO 3
DO 4 J=2,N
K=J-1
DO 5 L=1,K
IF(NOMAGR(K+2-L).CT.NOMAGR(K+1-L))GO TO 4
NOMPR=NOMAGR(K+1-L)
NOMAGR(K+1-L)=NOMAGR(K+2-L)
5 NOMAGR(K+2-L)=NOMPR
4 CONTINUE
DO 6 J=1,N
IF(NOMAGR(J).NE.NMAGR1(J)) GO TO 3
6 CONTINUE
GO TO 7
3 IF(PRIZNK.NE.0)GO TO 8
IF(NOMZAP.NE.20*MASNST(I)+1)GO TO 9
NOMZAP=NOMZAP-1
9 NOMHAR=NOMZAP-20*(MASNST(I)-1)
WRITE(3,10) MASNST(I),NOMHAR
10 FORMAT(/1H ,Т23,32СОСТАВ ОБОРУДОВАНИЯ НА СТАЦИИ N,12,30H НЕ НАЙ
СДЕН,КОРРЕКТИРОВАТЬ ПОД НОМЕРОМ ,12)
IND=IND+1

```

```

14 IF (I.EQ.KOLST) GO TO 11
   I=I+1
   GO TO 12
7  NOMZAP=NOMZAP-1
   READ(6,NOMZAP) PRIZNK, N, (NDMACR(J), J=1, N), KOLTCH, MINSUM, MAXSUM,
   CSHAGY, (HARRAS(J), J=1, KOLTCH)

   MASUCH(I)=KOLTCH-1
   J=1
   HARRST(J,I)=MINSUM
   L=J+2
   HARRST(L,I)=HARRAS(J)
13 J=J+1
   L=J+2
   IF (J.EQ.KOLTCH) GO TO 19
   HARRST(J+2-I,I)=MINSUM+1.71*CSHAGY
   HARRST(L,I)=HARRAS(J)
   GO TO 13
19 HARRST(J+2-I,I)=MAXSUM
   HARRST(L,I)=HARRAS(J)
   GO TO 14
11 IF (IND.NE.0) GO TO 16
   READ(1,15) SHAGY, MSHAGY, SHAGP, MSHAGP, EDIZM, PSUMZD
15 FORMAT(4E9.3/4A4/E12.3)
   CALL DYNPRG(KOLST, MASUCH, KOLUCH, HARRST, SHAGY, MSHAGY, SHAGP, MSHAGP,
   COTVET, KOLZNY, HAR)
   CALL ZATROP(PSUMZD, HARRST, EOPT)
   CALL MOSHOP(PSUMZD, HARRST, HAR, OTVET)
   CALL VIVOD(EDIZM, TEXT, OTVET, MASNST)
18 STOP
16 WRITE(3,17)
17 FORMAT(//1M, T26.6RHHY ПОЛНИТЬ КОРРЕКЦИЮ ХАРАКТЕРИСТИК СТАНЦИЙ С ПО-
   СМОЩЬЮ ПРОГРАММЫ СОСНАР)
   GO TO 18
END

```

Подпрограмма DYNPRG

=====

```

SUBROUTINE DYNPRG(N, A, KOLUCH, IZLUM, SHAGY, MSHAGY, SHAGP, MSHAGP, COTVET,
C, KOLZNY, HAR)
  DIMENSION EDIZM(4)
  INTEGER N*4, A*2(N), OTVET*4(N), NDMACR*2(N), K*2, TEXTOR*2(4),
  PECHAT*2(225)
  DATA PECHAT /('T', '21', ' ', 'I', 'X', ' ', '11', 'H', '5', '-', ' ', ' ', ' ', ' ', '7',
  C1H '-', '3', '-', ' ', ' ', ' ', '6M', 'J', '-', ' ', ' ', 'T', '21', ' ', ' ', ' ', 'H', '1', ' ', '10', 'X', ' ',
  C11H', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', 'T', '21', ' ', ' ', ' ', '2', ' ', 'H1',
  C1', ' ', 'H', 'OM', 'EP', ' ', ' ', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', 'T', '21',
  C1', ' ', '1H', 'I', ' ', '10', 'X', ' ', '1H', 'I', ' ', ' ', ' ', '62', 'X', ' ', 'I2', ' ', '3', 'H',
  C1', 'I', ' ', ' ', 'T2', ' ', ' ', ' ', '12', 'H1', 'A', ' ', 'PE', ' ', 'GA', 'TA', 'I', ' ', ' ', 'I',
  C1', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', ' ', 'T', '21', ' ', ' ', ' ', '10', 'X', ' ',
  C11H', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', ' ', 'T', '21', ' ', ' ', ' ', '1H', 'I', ' ', '11',
  C1', 'H', '5', '-', ' ', 'I', ' ', ' ', ' ', '7', 'H', '3', '-', ' ', ' ', ' ', 'T', '21', ' ', ' ', ' ', '1H',
  C1', ' ', '10', 'X', ' ', '1H', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', 'T', '21', ' ', ' ',
  C12H', 'I', ' ', 'HA', 'EP', '3', 'X', 'A', ' ', ' ', ' ', '6', 'X', ' ', '1H',
  C1I', ' ', 'T', '21', ' ', ' ', '1H', 'I', ' ', ' ', ' ', '7H', 'I', ' ', ' ', ' ', '11', 'X',
  C1I4', '2', 'H', 'I', ' ', ' ', ' ', '21', ' ', ' ', '2H', 'I', ' ', ' ', 'H', '18T',
  C1', ' ', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', 'T', '21', ' ', ' ', ' ', '1H',
  C1I', '10', 'X', ' ', '1H', 'I', ' ', ' ', ' ', '6', 'X', ' ', '1H', 'I', ' ', ' ', 'T', '22',
  C1', ' ', '11', 'H', '5', '-', ' ', ' ', ' ', 'T', 'H', '3', '-', ' ', ' ', ' ', '6M', 'J', '-',
  C1', ' '
  REAL*4 IZLUM(KOLUCH, N), SHAGY, MSHAGY, SHAGP, MSHAGP, HAR(KOLZNY, N),
  CB(200), BPR(200)

```

BLOCK N1

```

I=2*A(1)+1
HAR(1,1)=IZLON(1,1)
J=1
PROMSH=0
DO 5 K=1,M
5 PROMSH=PROMSH+IZLON(2*A(K)+1,K)
SHAGY=PROMSH*SHAGY/100
IF(SHAGY-MSHAGY)74,74,6
6 SHAGY=MSHAGY
C
74 IF(HAR(J,1)+SHAGY-IZLON(1,1))8,9,9
8 HAR(J+1,1)=HAR(J,1)+SHAGY
J=J+1
GO TO 74
9 HAR(J+1,1)=IZLON(1,1)
K=1
I=A(1)
Y=IZLON(1,1)
J=1
10 DELTAB=(IZLON(2*J+2,1)-IZLON(2*J,1))/(IZLON(2*J+1,1)-IZLON(2*J-1,1)
C)
B(K)=IZLON(2*J,1)+DELTAB*(Y-IZLON(2*J-1,1))
100 K=K+1
Y=Y+SHAGY
IF(Y-IZLON(2*J+1,1))72,72,73
72 B(K)=B(K-1)+DELTAB*SHAGY
GO TO 100
73 IF(J-1)103,104,104
103 J=J+1
DO 7 L=J+1
IF(Y-IZLON(2*L+1,1))101,101,7
101 J=L
IF(L-1)102,104,104
7 CONTINUE
104 B(K)=IZLON(2*L+2,1)
KONETS=K

```

BLOCK N3

```

C
PROTS=SHAGP
SHAGP=0
DO 11 I=1,N
PROMSH=IZLON(2*A(I)+1,I)*PROTS/100
IF(PROMSH-MSHAGP)13,12,12
13 IF(SHAGP-PROMSH)14,11,11
14 SHAGP=PROMSH
11 CONTINUE
12 SHAGP=MSHAGP

```

BLOCK N4

C M=2

BLOCK N5

```

C
MIN=IZLON(1,1)+IZLON(1,2)
MAX=IZLON(2*A(1)+1,1)+IZLON(2*A(2)+1,2)
X=IZLON(1,1)
Z=IZLON(2*A(1)+1,1)

```

BLOCK N6

```

C
40 MT=IZLON(1,M)
MT=IZLON(2*A(M)+1,M)
Y=MIN
L=1
BPR(1)=0
DO 15 I=1,M
15 BPR(I)=BPR(I)+IZLON(2,1)
HAR(1,M)=IZLON(1,M)
K=1

```

```

      BMXSUM=0
      DO 18 I=1,M
18    BMXSUM=BMXSUM+IZLDM(2*A(I)+2,I)
36    K=K+1
      Y=Y+SHAGY
      IF(Y-MAA)16,17,17
16    BOPT=BMXSUM
      PT=TMF
      SCHNA2=1
22    IF(PT-RMT)19,20,20
19    IF(Y-PT-Z)21,21,105

21    IF(Y-PT-X)23,24,24
24    J=A(M)+1
      DO 25 I=2,J
      IF(IZLDM(2*I-1,M)-PT)25,26,26
25    CONTINUE
26    DP=IZLDM(2*I-1,M)-IZLDM(2*(I-1)-1,M)
      DB=IZLDM(J+1,M)-IZLDM(2*(I-1),M)
      DFT=DB/DP*(PI-IZLDM(2*(I-1)-1,M))
      FT=IZLDM(2*(I-1),M)+DFT
      IF(Y-PT-Z)107,106,106
106    J=M-1
      BYPT=0
      DO 108 I=1,J
108    BYPT=BYPT+IZLDM(2*A(I)+2,I)
      GO TO 109
107    DYPT=Y-PT-X
      I=INT(DYPT/SHAGY)
      IF(I+2-KONETS)27,28,28
27    PROMP=(I+1)*SHAGY
      GO TO 29
28    PROMP=Z-X

29    BYPT=B(I+1)+(B(I+2)-B(I+1))/(PROMP-I*SHAGY)+(DYPT-I*SHAGY)
109    BTEK=FT+BYPT
      IF(BTEK-BOPT)30,30,105
30    BOPT=BTEK
      PTOPT=PT
105    PT=PT+SHAGP
      GO TO 22
20    IF(Y-RMT-X)23,31,31
31    IF(SCHNA2-1)33,33,23
33    SCHNA2=2
      PT=RMT
      GO TO 24
23    L=L+1
      HAR(L,M)=PTOPT
      BPR(K)=BOPT
      GO TO 36
17    HAR(K,M)=RMT
      BPR(K)=BMXSUM

C
      DO 37 I=1,K
37    B(I)=BPR(I)
      KONETS=K

C
      M=M+1

```

БЛОК N7

БЛОК N8

```

C                                     БЛОК N9
      IF(M-N)38,38,39
C                                     БЛОК N10
38 MIN=MIN+IZLDM(1,M)
   MAX=MAX+IZLDM(2*A(M)+1,M)
   X=X+IZLDM(1,M-1)
   Z=Z+IZLDM(2*A(M-1)+1,M-1)
   GO TO 40
C                                     БЛОК N11
39 RETURN
C                                     ТОЧКА ВХОДА ЗАТРОП
   ENTRY ЗАТРОП(PSUMZD,IZLDM,BOPT)
C                                     БЛОК N12
   XNACHY=IZLDM(1,1)
   DO 42 I=2,N
42  XNACHY=XNACHY+IZLDM(1,I)
   K=INT((PSUMZD-XNACHY)/SHAGY)+1
C                                     БЛОК N13
   BOPT=(B(K+1)-B(K))/SHAGY*(PSUMZD-XNACHY-(K-1)*SHAGY)+B(K)
C                                     БЛОК N14
   RETURN
C                                     ТОЧКА ВХОДА МОШОП
   ENTRY МОШОП(PSUMZD,IZLDM,HAR,OTVET)
C                                     БЛОК N15
   M=N
C                                     БЛОК N16
45 L=INT((PSUMZD-XNACHY)/SHAGY)+1
   PTOPT=(HAR(L+1,M)-HAR(L,M))/SHAGY*(PSUMZD-XNACHY-(L-1)*SHAGY)+
CHAR(L,M)
C                                     БЛОК N17
   OTVET(M)-PTOPT
   XNACHY=XNACHY-IZLDM(1,M)
C                                     БЛОК N18
   M=M-1
C                                     БЛОК N19
   IF(M)44,44,43
C                                     БЛОК N20
43 PSUMZD=PSUMZD-PTOPT
   GO TO 45
C                                     БЛОК N21
44 RETURN
C                                     ТОЧКА ВХОДА ВИВОД
   ENTRY ВИВОД(EDIZM,TEXFOR,OTVET,NOMAGR)
C                                     БЛОК N22
   DO 62 I=1,4
62 PECHAT(74+I)=TEXFOR(I)
C                                     БЛОК N23
   WRITE(3,46)BOPT,EDIZM
46  FORMAT(5('/),' ',147,'РАСХОД ТОПЛИВА=',F12,3,5A4)
   WRITE(3,201)
201 FORMAT(5('/))
C                                     БЛОК N24
   STROK=N
   STROK=STROK/10
   IF(STROK-AINT(STROK))48,48,49
48 I=AINTE(STROK)
   GO TO 50
49 I=AINTE(STROK)+1
C                                     БЛОК N25
50 IF(N-10)51,52,52

```

C	51 K=N GO TO 53	БЛОК N26
C	52 K=10	БЛОК N27
C	53 J=1	БЛОК N28
C	54 IF(K-10)56,57,57	БЛОК N29
C	56 ISIMVL=K+16624 PECHAT(13)=ISIMVL-1 PECHAT(33)=ISIMVL PECHAT(49)=ISIMVL PECHAT(63)=ISIMVL PECHAT(82)=ISIMVL PECHAT(96)=ISIMVL PECHAT(114)=ISIMVL PECHAT(130)=ISIMVL PECHAT(146)=ISIMVL PECHAT(160)=ISIMVL PECHAT(179)=ISIMVL PECHAT(193)=ISIMVL PECHAT(210)=ISIMVL-1 GO TO 58	БЛОК N30
C	57 PECHAT(13)=16633 PECHAT(33)=-3600 PECHAT(49)=-3600 PECHAT(63)=-3600 PECHAT(82)=-3600 PECHAT(96)=-3600 PECHAT(114)=-3600 PECHAT(130)=-3600 PECHAT(146)=-3600 PECHAT(160)=-3600 PECHAT(179)=-3600 PECHAT(193)=-3600 PECHAT(210)=16633	БЛОК N31
C	58 L=J+K-1	БЛОК N32
C	WRITE(3,PECHAT)(NOMAGR(M),M=J,L).(OTVET(M),M=J,L)	БЛОК N33
C	I=I-1	БЛОК N34
C	IF(I)60,60,59	БЛОК N35
C	59 J=K+1	БЛОК N36
C	K=N-L IF(K.GT.10)K=10 GO TO 54	БЛОК N37
C	60 WRITE(3,01) 61 FORMAT(5(' '),T45,'ПРИВЕТ СИВОМУ ! (В. М. ШУКШИН)') RETURN END	БЛОК N38

Подпрограмма HARST

=====

```

SUBROUTINE HARST(NOMHAR)
  INTEGER*2 KOLST, EDIZM(8), UPRCAR(19), ETALON(57) / ТОЛЬКО ОТПЕЧАТАТЬ
  ХАРАКТЕРИСТИКИ ТОЛЬКО ЗАПОМНИТЬ ХАРАКТЕРИСТИКИ ЗАПОМНИТ
  СБ И ОТПЕЧАТАТЬ ХАРАКТЕРИСТИКИ / A(20), NOMAGR(20), OTVET=4(20),
  CPRIZNK / 00 / N=4
  REAL SHAGY, MSHAGY, SHAGP, MSHAGP, IZLON(100, 20), HAR(200, 20), HARRAS(50
  C), MINSUM, MAXSUM
  DATA KOLUCH / 100 /, KOLZNY / 200 /
  READ(1, 1) KOLST
1  FORMAT(13)
  NOMST=1
  ENTRY HARST1(KOLST, NOMST, NOMHAR)
  READ(1, 2) EDIZM, SHAGY, MSHAGY, SHAGP, MSHAGP, UPRCAR
2  FORMAT(8A2/4E9.2/19A2)
  DO 3 I=1, 19
  IF(UPRCAR(I), NE, ETALON(I)) GO TO 4
3  CONTINUE
  GO TO 5
4  DEFINE FILE 6(1000, 262, L, NOMZAP)
5  READ(1, 6) N, (NOMAGR(I) - A(I), I=1, N)
6  FORMAT(13/24(13))
  DO 7 I=1, N
  K=2*(A(I)+1)
  READ(1, 8) (IZLON(J, I), J=1, K)
8  FORMAT(3(E7.0, E15.3))
7  CONTINUE
  MINSUM=IZLON(1, 1)
  DO 9 I=2, N
9  MINSUM=MINSUM+IZLON(1, I)

  MAXSUM=IZLON(2*(A(I)+1), I)
  DO 10 I=2, N
10 MAXSUM=MAXSUM+IZLON(2*(A(I)+1), I)
  J=1
  I=1
  TEKP=MINSUM
  CALL DYNPRG(N, A, KOLUCH, IZLON, SHAGY, MSHAGY, SHAGP, MSHAGP, OTVET, KOL
  C ZNY, KAR)
11 CALL ZATROP(TEKP, IZLON, BOPT)
  HARRAS(I)=BOPT
  SHAGY=AINT((MAXSUM-MINSUM)/49)+1
  TEKP=TEKP+SHAGY
  I=I+1
  IF(TEKP-MAXSUM) 11, 12, 12
12 KOLTCH=1
  BMSUM=IZLON(2*(A(J)+1), 1)
  DO 13 J=2, N
13 BMSUM=BMSUM+IZLON(2*(A(J)+1), J)
  HARRAS(I)=BMSUM
  DO 14 I=1, 19
  IF(UPRCAR(I), NE, ETALON(19+I)) GO TO 15
14 CONTINUE
  GO TO 16
15 DO 17 I=1, 19
  IF(UPRCAR(I), NE, ETALON(38+I)) GO TO 18
17 CONTINUE
  CALL PECHAT(HARRAS, KOLTCH, MINSUM, MAXSUM, SHAGY, NOMST, NOMAGR, N, EDIZM
  C)
  GO TO 16

```

```

18 CALL PECHAT (HARRAS,KOLTCH,MINSUM,MAXSUM,SHAGY,NOMST,NOMAGR,N,EDIZM
C)
GO TO 19
16 DO 21 I=2,N
K=I-1
DO 22 J=1,K
IF (NOMAGR(K+2-J).GT.NOMAGR(K+1-J)) GO TO 21
NOMPR=NOMAGR(K+1-J)
NOMAGR(K+1-J)=NOMAGR(K+2-J)
22 NOMAGR(K+2-J)=NOMPR
21 CONTINUE
WRITE (6,'2#*(NOMST-1)+NOMHAR)PRIZNK,N,(NOMAGR(I),I=1,N),KOLTCH,MIN
CSUM,MAXSUM,SHAGY,(HARRAS(I),I=1,KOLTCH)
19 NOMST=NOMST+1
IF (NOMST-KOLST)5,5,2#
2# RETURN
END

```

Подпрограмма PECHAT

=====

```

SUBROUTINE PECHAT(MASZNY,KOLZNY,XMIN,XMAX,SHAGX,NOMST,NOMAGR,N,
CEDIZMY)
INTEGER*2 N=4,NOMST*4,NOMAGR(N),VIVOD1(55),VIVOD2(228),KOLZNY=4
REAL SHAGX,XMAX,XMIN,MASZNY(KOLZNY),NAGRST(1#)
DIMENSION EDIZMY(4)
DATA VIVOD1/'(2(//))1H ,T42,34HPACXOAHNA XAPAKTEPCTHKA CTAHUHH N,
CI2/2(//),1H ,T16,3#HVKACHENY B PABOTY ACPERATY MN:2#(I3)) '//
DATA VIVOD2/'(//2(//))//T//2//18//H//8*--//)//
C'(1//0H//5*--//)//T2//1//HI//1//6X//1//HI//
C//9//X//1H//I//1T//2//1H//I//4X//8//H//AГ//PY//
C'3X//A//4X//1//HI//9//X//1H//I//T//2//
C'1H//I//16//X//1H//I//E9//3//
C'1H//I//T//2//1H//I//16//H//CT//AH//UHH//M//
C'MB//T//1H//I//9//X//1H//I//T//
C'2//1H//I//16//X//1H//I//9//X//1H//I//T//
C'2//18//H//8*--//1//0H//5*--//T2//1//
C'HI//1//6X//1//HI//9//X//1H//I//T//2//
C'1H//I//16//H//PA//CX//OD//T//OH//MH//BA//1H//
C'I//E//9//3//1H//I//T//2//1H//I//16//X//1H//
C//1H//I//9//X//1H//I//T//2//1H//I//4A//6//
C'X//1H//I//9//X//1H//I//T//2//1H//I//16//
C//1//1//H//5*--//)//
WRITE (3,VIVOD1)NOMST,NOMAGR
DO1 I=1,1#
1 NAGRST(I)=#
STOK=KOLZNY
STOK=STOK/1#
IF (STOK-AINT (STOK))2,2,3
2 I=AINT (STOK)
GO TO 4
3 I=AINT (STOK)+1
4 IF (KOLZNY-1#)5,6,6
5 K=KOLZNY
GO TO 7
6 K=1#
7 J=1
8 NAGRST(1)=XMIN+SHAGX*(J-1)
IF (K-1)1#,1#,11
11 DO 9 L=2,K

```

```

9  NAGRST(L)=NAGRST(L-1)+SHAGX
  IF(NAGRST(L).GT.XMAX)NAGRST(L)=XMAX
  IF(K-1)10,12,12
10 ISIMVL=K+16624
  VIVOD2(17)=ISIMVL
  VIVOD2(34)=ISIMVL
  VIVOD2(161)=ISIMVL
  VIVOD2(54)=ISIMVL
  VIVOD2(67)=ISIMVL
  VIVOD2(92)=ISIMVL
  VIVOD2(105)=ISIMVL
  VIVOD2(123)=ISIMVL
  VIVOD2(140)=ISIMVL
  VIVOD2(174)=ISIMVL
  VIVOD2(189)=ISIMVL
  VIVOD2(202)=ISIMVL
  VIVOD2(220)=ISIMVL
  GO TO 13
12 VIVOD2(17)=-3600
  VIVOD2(34)=-3600
  VIVOD2(54)=-3600
  VIVOD2(67)=-3600
  VIVOD2(92)=-3600
  VIVOD2(105)=-3600
  VIVOD2(123)=-3600
  VIVOD2(140)=-3600
  VIVOD2(161)=-3600
  VIVOD2(174)=-3600
  VIVOD2(189)=-3600
  VIVOD2(202)=-3600
  VIVOD2(220)=-3600
13 L=J+K-1
  WRITE(3,VIVOD2)(NAGRST(M),M=1,K),(MASZNY(M),M=J,L),EDIZNY
  I=I-1
  IF(I)15,15,14
14 J=L+1
  K=KOLZNY-L
  IF(K.GT.10)K=10
  GO TO 8
15 RETURN
  END

```

СОДЕРЖАНИЕ

Предисловие	3
Лабораторная работа 1. Применение метода динамического программирования к оптимизации распределения активных мощностей между агрегатами электростанции	5
Лабораторная работа 2. Изучение возможностей, предоставляемых пользователю операционной системой ДОС/ЕС на примере задачи оптимизации распределения нагрузок между агрегатами электростанции.	34
Лабораторная работа 3. Организация корректируемого файла расходных характеристик электростанций для использования при оперативной оптимизации режима энергосистемы	52
Лабораторная работа 4. Коррекция файла расходных характеристик электростанций при оперативной оптимизации режима энергосистемы	86
Лабораторная работа 5. Подготовка комплекса программ оперативной оптимизации режима энергосистемы по активной мощности и эксплуатации и оптимальное распределение мощностей между электростанциями	110
Лабораторная работа 6. Организация эксплуатации, пополнения и обновления комплекса программ на примере задачи оптимизации режима энергосистемы по активной мощности	127
Литература	135
Приложение	136

*Станислав Константинович Гурский,
Сергей Васильевич Домников,
Олег Игоревич Александров*

**Оптимизация режимов работы энергосистем.
Лабораторный практикум**

Зав. редакцией Л.Д.Духвалов
Редактор Н.М.Патышева
Худож.редактор Ю.С.Сергачев
Техн. редактор Л.И.Счисленок
Мл.редактор М.С.Молчанова,
Корректоры В.П.Шкредова, И.И.Тарасик
Оператор М.К.Борисова

ИБ № 1920

Подписано в печать 08. 07. 85 г. АТ 17340. Формат
60х90 1/16. Бумага офсет. Гарнитура Пресс Роман.
Усл.печ.л. 9,25. Усл. кр.-отт. 9,625. Уч.-изд.л. 9,93 .
Тираж 1300 экз. Зак. 6100. Цена 40 к.

Издательство "Вышэйшая школа" Государствен-
ного комитета БССР по делам издательств, поли-
графии и книжной торговли. 220048, Минск,
проспект Машерова, 11.

Типография "Победа", 222310, Молодечно,
ул. Тавлая, 11.

Отпечатано с оригинал-макета, подготовленного
в издательстве "Вышэйшая школа".