

**Московский государственный технический университет им. Н.Э.Баумана**

**С.В.Родионов, М.В. Москина**

# **Методическое руководство**

**для практических занятий по курсам:**

**"Основы автоматизированного проектирования" и "Исследование операций"**

## **ПРОЕКТИРОВАНИЕ ОПТИМАЛЬНЫХ РАСПИСАНИЙ**

г. Москва 2002г.

## АННОТАЦИЯ

Настоящее руководство содержит сведения необходимые для практического изучения проблемы составления оптимальных конвейерных расписаний. Эта проблема может быть актуальна на этапе технологической подготовки производства при выборе очередности последовательной обработки заданной партии деталей на конечном множестве станков, образующих систему многофазного конвейера для выполнения требуемого маршрута технологических операций. Другое важное приложение этой проблематики связано с организацией конвейерной обработки данных в многопроцессорных системах, когда необходимо определить порядок обслуживания пакета вычислительных заданий, операции которых последовательно выполняются отдельными процессорами.

В обоих случаях целесообразно составить оптимальное расписание, при котором суммарная продолжительность обработки партии деталей или пакета заданий будет минимальна. Составление оптимальных расписаний обеспечивают проектные процедуры структурного синтеза, основанные на формальных моделях и методах теории расписаний.

Данное руководство состоит из 2-х частей. В теоретической части рассмотрены постановка задачи и методы проектирования оптимальных расписаний в 2-х фазной конвейерной системе обслуживания, состоящей из 2-х обслуживающих устройств (машин). Эту классическую проблему теории расписаний обычно называют задачей 2-х машин. В практической части руководства приведено описание программно-методического комплекса SHD2, который может быть использован для автоматизации проектирования оптимальных расписаний в условиях задачи 2-х машин под управлением операционной системы UNIX (Linux, Free BSD, Sun OS, Solaris, Ultrix).

## СОДЕРЖАНИЕ

### 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1. Основные понятия теории расписаний
2. Задача двух машин
3. Перестановочный прием
4. Метод полного перебора
5. Анализ расписаний

### 2. ПРАКТИЧЕСКАЯ ЧАСТЬ

1. Техническое задание
2. Автоматизация составления расписаний
3. Порядок выполнения задания
4. Организация самоконтроля

### 3. СПИСОК ЛИТЕРАТУРЫ

## ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

### Основные понятия теории расписаний

Терминологическую базу теории расписаний составляют следующие категории: операция, работа, машина. Физическая природа этих понятий безразлична для теории расписаний. Существенны только их общие свойства, инвариантные к прикладному содержанию.

Операция - элементарное действие, подлежащее выполнению. Каждая операция количественно характеризуется длительностью выполнения, принадлежит определенной работе и реализуется на соответствующей машине.

Работа - целенаправленная последовательность операций. Очередность операций каждой работы строго фиксирована, формально задается отношением порядка и устанавливается, исходя из соответствующих технологических соображений прикладного характера. Множество работ, подлежащих выполнению, образует партию работ. Состав партии фиксирован до начала ее обработки.

Машина - устройство для выполнения операций работ. В общем случае, машина есть нескладируемый ресурс, который в ходе выполнения назначенной операции не расходуется, а производит сам некий расходуемый фактор, не подлежащий складированию, например, машино-смены. Множество машин, необходимое для выполнения операций работ заданной партии, составляет систему обслуживания.

Реализация выполнения операций партии работ машинами системы обслуживания называется процессом обслуживания. Процесс обслуживания характеризуется следующими показателями:

- число машин в системе обслуживания;
- количество операций в каждой работе;
- порядок прохождения машин для каждой работы.

По числу машин процессы обслуживания подразделяются на одномашинные, когда все операции работ выполняются 1-й машиной, и многомашинные, если в системе обслуживания есть несколько одно- или разнотипных машин. По количеству операций работ процессы обслуживания подразделяются на однофазные, когда каждая работа состоит из единственной операции, и многофазные, если в состав каждой работы входит несколько операций. По порядку прохождения машин процессы обслуживания подразделяются на конвейерные, когда порядок прохождения машин одинаков для всех работ партии, и произвольные, если порядок прохождения машин различен или не регламентируется.

Для реализации процесса обслуживания необходимо составить расписание, которое определяет сроки выполнения операций для каждой машины и (или) очередность работ партии. Формальные модели и методы планирования

оптимальных расписаний изучает теория расписаний. В настоящее время в ней реализован достаточно узкий класс формальных моделей и методов планирования простого процесса обслуживания, для которого существенны следующие ограничения:

- параллельное выполнение операций одной работы не допустимо;
- каждая операция выполняется полностью одной машиной;
- прерывания выполнения операций отсутствуют;
- любая машина в любой момент выполняет не более одной операции.

В качестве критерия оптимальности расписания могут быть приняты либо суммарная длительность процесса обслуживания заданной партии работ, либо среднее время прохождения работ в системе обслуживания.

В теории расписаний рассматривается большое число модельных задач оптимального планирования простого процесса обслуживания. Их формальные постановки и численные методы решения определяются выбором критерия оптимальности и показателями процесса обслуживания. Одной из наиболее хорошо изученных задач теории расписаний является задача 2-х машин.

## **Задача двух машин**

В этой задаче рассматривается проблема составления оптимального конвейерного расписания простого 2-х фазного процесса обслуживания в системе обслуживания из 2-х машин для произвольной по объему, но фиксированной во времени партии работ. Это означает, что все работы заданной партии состоят из 2-х операций, которые в одинаковом порядке выполняются парой машин системы обслуживания. Длительности операций всех работ и объем партии считаются заданными. Требуется определить порядок выполнения работ и сроки начала их операций, при которых достигается минимальная продолжительность простого процесса обслуживания.

Классической прикладной интерпретацией задачи 2-х машин является составление оптимального расписания конвейерной обработки партии деталей на 2-х станках или пакета заданий 2-мя процессорами. По этой причине эту задачу часто называют задачей о 2-х станках или задачей о 2-х процессорах. Независимо от прикладной интерпретации исходными данными задачи 2-х машин являются:

- состав партии работ;
- перечень машин системы обслуживания;
- продолжительности выполнения операций.

Состав партии формально задает конечное множество:

$$W = \{ W_q, q = 1, \dots, n \},$$

где  $W_q$  - обозначение  $q$ -й работы, а  $n$  - число работ партии. Например, для обозначения работ могут быть использованы строчные буквы латинского алфавита:

$$W = \{ a, b, c, \dots x, y, z \}.$$

Систему обслуживания задает упорядоченная пара  $(A, B)$ , где  $A$  и  $B$  обозначают машины для выполнения 1-й и 2-й операции каждой работы. Длительности операций каждой работы  $W_q$  на машинах  $A$  и  $B$  задаются величинами  $A_q$  и  $B_q$  ( $q = 1, \dots, n$ ), соответственно. Причем,  $A_q = 0$  или  $B_q = 0$ , если работа  $W_q$  не обслуживается машиной  $A$  или  $B$ , т.е. имеет только одну операцию.

Допустимые решения задачи 2-х машин образуют множество расписаний, определяющих сроки операций в условиях простого процесса обслуживания. Любое допустимое расписание для задачи 2-х машин однозначно определяет очередность выполнения работ. Такие расписания относятся к классу перестановочных, т.к. могут быть формально заданы перестановкой работ:

$$S = ( [1], [2], \dots [i], \dots [n] ),$$

где  $[i]$  обозначает работу, которая имеет  $i$ -ю очередность обслуживания.

Если  $i$ -ю очередь имеет работа  $W_q$ , то длительности ее операций  $A_q$  и  $B_q$  обозначаются  $A[i]$  и  $B[i]$ , соответственно. Для партии объемом  $n$  работ можно построить  $n!$  различных перестановок работ, определяющих  $n!$  различных допустимых расписаний.

Специфика задачи 2-х машин позволяет ограничиться рассмотрением только перестановочных расписаний, в которых машина  $A$  не имеет простоев. Желаемую плотность упаковки графика машины  $A$  можно обеспечить сдвигом даты начала всех операций влево. Однако, упаковка графика машины  $A$  не означает отсутствие простоев машины  $B$ . Более того, в расписании обычно необходимо предусмотреть простои машины  $B$ , чтобы гарантировать выполнение условий простого процесса обслуживания.

Пусть  $X[i]$  обозначает интервал простоя машины  $B$  непосредственно перед выполнением на ней работы  $[i]$ . Для гарантии выполнения условий простого процесса обслуживания интервалы локальных простоев машины  $B$  должны удовлетворять следующим рекуррентным соотношениям:

$$X[i] = \max(Y[i] - X[1] - X[2] - \dots - X[i-1], 0),$$

где величины  $Y[i]$  определяются как:

$$Y[i] = A[1] + A[2] + \dots + A[i] - B[1] - B[2] - \dots - B[i-1].$$

Пусть  $D[i]$  обозначает частную сумму простоев машины  $B$  до начала выполнения на ней работы  $[i]$ , т.е.:

$$D[i] = X[i] + X[2] + \dots + X[i].$$

Подстановка выражений локальных простоев машины В позволяет представить указанные частные суммы в виде следующих рекуррентных соотношений:

$$D[i] = \max(Y[i], D[i-1]), D[1] = A[1],$$

которые после алгебраических преобразований могут быть представлены следующим образом:

$$D[i] = \max(Y[i], Y[i-1], \dots, Y[1]).$$

Как следует из содержательной постановки задачи 2-х машин, качество расписания определяет суммарная продолжительность обслуживания партии работ. Для любой перестановки S работ партии общую продолжительность простоя процесса обслуживания устанавливает суммарная длительность операций и простоев машины В, количественную оценку которой дает следующая функция цели:

$$F(S) = B[1] + \dots + B[n] + \max(Y[n], Y[n-1], \dots, Y[1]).$$

Оптимальное расписание для задачи 2-х машин должно гарантировать минимальную продолжительность обслуживания партии. Формально оно определяется перестановкой работ S, которая обеспечивает минимум функции цели F(S) на множестве всех n! различных перестановок работ партии. Выполнение условий простоя процесса обслуживания гарантирует способ назначения простоев машины В, рассмотренный выше.

Следует отметить, что суммарная длительность операций машины В не зависит от порядка обслуживания работ партии, так как:

$$B[1] + \dots + B[i] + \dots + B[n] = \text{const.}$$

Это обстоятельство позволяет упростить функцию цели F(S) и свести задачу 2-х машин к выбору перестановки работ, которая минимизирует суммарный простой машины В, или обеспечивает минимум максимальной из величин Y[i] (i=1,...,n):

$$\max (Y[n], Y[n-1], \dots, Y[1]) \rightarrow \min$$

на множестве всех n! перестановок S работ партии.

Рассмотренная формальная модель позволяет отнести задачу 2-х машин к классу задач упорядочивания, в которых нужно найти перестановку, доставляющую экстремум функции цели на множестве допустимых перестановок. Оптимальное решение задач упорядочивания теоретически всегда может быть найдено методом полного перебора. Для ряда "простых" задач упорядочивания, к которым относится задача 2-х машин, оптимальное решение удается получить с помощью

специальных перестановочных приемов, позволяющих существенно сузить множество рассматриваемых вариантов по сравнению с их полным перебором.

## Перестановочный прием

Существующие эффективные методы построения оптимальных расписаний для задачи 2-х машин обеспечивают упорядочивание работ партии на основе перестановочного приема. Его суть заключается в предварительном исследовании влияния взаимного расположения соседних работ партии на качество расписания. Для каждой пары соседних работ перестановочный прием определяет необходимость их транспозиции, т.е. устанавливает, какая из пары работ должна выполняться первой.

Перестановочные приемы в задаче 2-х машин основаны на правиле Джонсона. Согласно этому правилу, для получения оптимального расписания достаточно, чтобы порядок обслуживания работ партии соответствовал следующему условию:

$$\min (A[i], B[i+1]) \leq \min (A[i+1], B[i]), i = 1, \dots, n-1.$$

Если для любых пар работ это условие является строгим неравенством, то существует единственное расписание, оптимальное по правилу Джонсона. Если для некоторых пар работ это условие обращается в равенство, то по правилу Джонсона можно построить более, чем 1 оптимальное расписание. Правило Джонсона удобно использовать для эффективной проверки качества готового расписания. Однако, его непосредственное применение для поиска оптимального расписания в задаче 2-х машин алгоритмически неудобно. Эффективное упорядочивание работ по правилу Джонсона обеспечивают два перестановочных приема, известные как алгоритм Джонсона и алгоритм приоритетов.

Алгоритм Джонсона разбивает процедуру синтеза оптимального расписания для партии из  $n$  работ на  $n$  последовательных шагов. На каждом шаге определяется место в расписании для одной из работ партии. При этом, на очередном шаге выбирается работа, которая обладает самой короткой, т.е. минимальной по длительности, операцией среди всех пока неразмещенных работ. Для размещения выбранной работы используется либо наименьшее, либо наибольшее по номеру свободное место расписания в зависимости от соотношения длительности ее операций на машинах  $A$  и  $B$ . Если короткая операция выбранной работы  $W_q$  выполняется машиной  $B$ , т.е.  $B_q < A_q$ , то для размещения работы  $W_q$  назначается наибольшее по номеру свободное место расписания. В остальных случаях ( $A_q < B_q$  или  $A_q = B_q$ ), выбранная работа  $W_q$  размещается на 1-м свободном месте расписания. На последнем шаге, оставшаяся неразмещенная работа назначается на единственное свободное место расписания.

При равенстве продолжительности коротких операций у 2-х или более работ возникает неоднозначность порядка их выбора и места размещения в расписании. Указанная неоднозначность может быть разрешена произвольным образом, порождая различные варианты оптимальных расписаний, которые отличаются



транспозицией работ с равными продолжительностями коротких операций. Оптимальные расписания, модифицированные путем такой транспозиции, продолжают удовлетворять правилу Джонсона.

Вычислительную сложность алгоритма Джонсона определяет суммарное число сравнений длительностей операций работ. На каждом шаге  $j = 0, \dots, n-1$  для выбора размещаемой работы нужно реализовать  $2 \cdot (n - j)$  сравнений, поэтому общее число сравнений за  $n$  шагов алгоритма равно  $n \cdot (n+1)$ . Таким образом, вычислительная сложность алгоритма полиномиально зависит от объема партии работ.

Алгоритм приоритетов разбивает процесс поиска оптимального расписания на 2 этапа. На 1-м этапе назначаются приоритеты всех работ партии. Приоритет каждой работы  $W_q$  ( $q = 1, \dots, n$ ) вычисляется по формуле:

$$P_q = \text{sign}(A_q - B_q) * \{ M - \min(A_q, B_q) \} .$$

Константа  $M$  в формуле приоритетов должна превосходить по величине продолжительность максимальной по длительности короткой операции среди всех коротких операций работ партии. Формально, выбор константы  $M$  должен быть ограничен снизу следующим неравенством:

$$M > \max \min (A_q, B_q),$$

где максимум ищется при всех  $q$  от 1 до  $n$ . На практике допустимо назначить константу  $M$  в формуле приоритетов следующим образом:

$$M = \max \min (A_q, B_q) + 1.$$

Таким образом, после 1-го этапа алгоритма приоритетов для каждой работы  $W_q$  ( $q = 1, \dots, n$ ) должен быть назначен приоритет  $P_q$  в диапазоне целых чисел от  $-M$  до  $M$ , то есть:

$$P_q = \leq | M |, q = 1, \dots, n .$$

На 2-м этапе алгоритма приоритетов выполняется сортировка работ партии с целью переставить работы в порядке неубывания их приоритетов. Для любой пары соседних работ  $[i]$  и  $[i+1]$  в полученной сортировкой перестановке  $S$  должно быть справедливо следующее неравенство:

$$P[i] = \leq P[i+1] , i = 1, \dots, n-1 .$$

После сортировки по приоритетам будет получено оптимальное расписание, удовлетворяющее правилу Джонсона. При равенстве приоритетов у 2-х или более соседних работ возможна вариация полученного оптимального расписания путем транспозиции работ с равными приоритетами. Для всех подобных вариаций оптимального расписания сохраняется справедливость правила Джонсона. В общем случае алгоритм приоритетов порождает более узкий класс оптимальных расписаний, чем алгоритм Джонсона. Любые "приоритетные" расписания могут

быть получены алгоритмом Джонсона после соответствующей модификации порядка выбора работ с равными продолжительностями коротких операций.

Вычислительную сложность алгоритма приоритетов определяет число сравнений приоритетов работ при сортировке на этапе 2. Известные методы сортировки обеспечивают полиномиальную зависимость числа сравнений от объема сортируемых данных, гарантируя вычислительную эффективность алгоритма приоритетов.

Таким образом, перестановочные приемы обеспечивают эффективное в вычислительном отношении упорядочивание работ партии для задачи 2-х машин, порождая класс оптимальных расписаний, удовлетворяющих правилу Джонсона.

## **Метод полного перебора**

Перестановочные приемы обеспечивают эффективный поиск оптимального решения задачи 2-х машин, т.к. вычислительная сложность реализующих их алгоритмов полиномиально зависит от объема партии работ. Однако, правило Джонсона, которое лежит в основе перестановочных приемов, устанавливает только достаточное условие получения оптимального расписания, которое не является необходимым. Поэтому, в общем случае, класс оптимальных расписаний для задачи 2-х машин может включать достаточно большое число расписаний, которые не удовлетворяют правилу Джонсона и, следовательно, недостижимы с помощью перестановочных приемов. Возможность перечисления всех расписаний в классе оптимальных обеспечивает полный перебор всех  $n!$  перестановок работ партии.

Существуют различные способы организации полного перебора перестановок: лексикографическое упорядочивание, циклический сдвиг, транспозиция смежных элементов. Для задачи 2-х машин удобно использовать вариант полного перебора, который порождает перестановки циклическим сдвигом, известный также как алгоритм вращения.

Естественный способ перечисления перестановок циклическим сдвигом состоит в том, что начав с некоторой произвольной перестановки, последовательно сдвигать по циклу на одно место влево все  $n$  работ партии. При каждом сдвиге 1-я работа текущей перестановки перемещается на последнее место без изменения взаимного расположения остальных, образуя новую перестановку. Такая организация циклического сдвига называется вращением. Вращение всех работ нужно продолжать, пока оно порождает новые перестановки, не встречавшиеся ранее. Перестановка считается оригинальной, когда после сдвига позиция последнего вращаемой части не равна его позиции в исходной перестановке. Если в результате очередного вращения получается ранее порожденная перестановка, нужно исследовать возможность построить оригинальную перестановку, применяя процедуру локального вращения последовательно для  $k = n-1, n-2, \dots, 2$  начальных работ при фиксированном положении остальных  $n - k$  хвостовых работ партии. Если локальное вращение первых  $1 < k < n$  работ порождает оригинальную

перестановку, следует продолжить вращение циклическим сдвигом всех  $n$  работ партии. В противном случае ( $k = 1$ ), перебор считается завершенным, т.к. перечислены все  $n!$  перестановок.

Для каждой оригинальной перестановки, порождаемой рассмотренным алгоритмом вращения, определяется продолжительность соответствующего расписания процесса обслуживания. Сравнительная оценка длительности позволяет отобрать в качестве оптимальных все перестановки с минимальной продолжительностью процесса обслуживания. Следует отметить, что полный перебор всех оптимальных решений задачи 2-х машин не может быть практически реализован для достаточно представительных партий работ по ресурсным соображениям, в связи с тем, что мощность перебора возрастает неполиномиально при увеличении объема партии и быстро превосходит возможности современных вычислительных систем.

## Анализ расписаний

Анализ расписания связан с оценкой ресурсных характеристик процесса обслуживания таких как общая продолжительность процесса обслуживания, суммарные длительности периодов простоя и функционирования машин, длина очереди накопления работ между машинами. Распространенными формами анализа расписаний являются диаграмма Ганта и имитация процесса обслуживания в режиме модельного времени.

Диаграмма Ганта - двумерная форма графического представления расписания в системе координат: время - номер машины. Она имеет смысл графика загрузки машин операциями работ партии. Для любого момента времени от начала до конца процесса обслуживания этот график позволяет однозначно определить операцию какой работы выполняет каждая машина и установить периоды простоя машин. Операции работ представляют горизонтальные отрезки, равные по длине продолжительностям выполнения операций в принятом масштабе времени. Вертикальное смещение каждого отрезка соответствует машине, которая обслуживает операцию, отображаемую им. Таким образом, количество горизонтальных уровней диаграммы Ганта равно числу машин системы обслуживания. Для задачи 2-х машин диаграмма Ганта состоит из 2-х уровней, которые соответствуют машинам А и В. Общее число горизонтальных отрезков на всех уровнях равно суммарному числу операций работ партии и простоев машин.

Чтобы отличать операции различных работ, отрезки операций диаграммы Ганта должны быть помечены сообразно обозначениям соответствующих работ. Например, если работы партии обозначить строчными литерами латинского алфавита, то каждый единичный интервал отрезка любой операции можно маркировать литерой работы, которой принадлежит эта операция. Для обозначения единичных интервалов простоя машин может быть использован символ тире (-). Таким образом, в рассмотренном символическом варианте диаграммы Ганта горизонтальные цепочки одинаковых литер соответствуют операциям работ партии или промежуткам простоя машин. Продолжительность

операций работ, промежутков простоя и процесса обслуживания в целом определяется соответствующим числом знакомест диаграммы Ганта. Следующий рисунок иллюстрирует диаграмму Ганта для расписания процесса обслуживания партии из 5-ти работ { a, b, c, d, e }, выполняемых в алфавитном порядке:

**Машина А: aabbccddddddeeee--**

**Машина В: --aaaaabbbc--ddd-ee**

Из приведенной диаграммы Ганта легко найти продолжительность процесса обслуживания (19 единиц) и суммарный простой машины В (5 единиц), величины которых определяют качество расписания. Для любого момента времени можно установить характер загрузки системы обслуживания операциями работ. Например, через 12 единичных интервалов после начала процесса обслуживания, машина А будет выполнять работу d, а машина В будет находиться в состоянии простоя (-).

Кроме анализа временных характеристик для ряда практических приложений целесообразна оценка длины очереди накопления работ между машинами в процессе обслуживания. Накопление работ между машинами происходит, когда после завершения операции некой работы на машине А, она не может быть мгновенно передана на машину В, занятую операциями предыдущих работ партии, что приводит к образованию очереди полуобслуженных работ на входе машины В.

Эффект накопления работ на между машинами можно наблюдать по диаграмме Ганта, приведенной выше. Из него видно, что после завершения операций работ b и c на машине А они не могут быть мгновенно переданы машине В, занятой обслуживанием предыдущей работы (a), образуя очередь на входе машины В. Максимальная длина очереди равна 2 работы в момент завершения обслуживания работы (a) машиной В.

Накопление работ между машинами не влияет на временные характеристики расписания. Однако, оценка максимальной длины межмашинной очереди работ необходима для оценки объема промежуточного буфера хранения работ между машинами при реализации процессов обслуживания, критичных по данному ресурсу. Если существует несколько оптимальных расписаний, равных по длительности процесса обслуживания, целесообразно выбрать расписание, реализация которого требуется минимальный объем промежуточного буфера для хранения очереди работ между машинами системы обслуживания.

# ПРАКТИЧЕСКАЯ ЧАСТЬ

## Техническое задание

Требуется составить и исследовать оптимальные расписания простого процесса обслуживания фиксированной партии работ в 2-х фазной конвейерной системе из 2-х машин А и В. Критерием оптимальности расписания считается общая продолжительность процесса обслуживания (дата завершения последней работы партии на машине В). Для выполнения технического задания необходимо:

- получить оптимальные перестановки работ партии, при которых суммарная продолжительность процесса обслуживания будет минимальна;
- построить график оптимального расписания в форме диаграммы Ганта;
- определить общую длительность оптимального расписания и суммарную продолжительность простоя машины В;
- оценить максимальную длину очереди накопления работ между машинами в процессе обслуживания по оптимальному расписанию.

Исходными данными для проектирования считаются состав партии работ и длительности операций работ на машинах системы обслуживания, которые заданы следующей таблицей:

Работа	a	b	c	d	e	f
Машина А	2	6	8	4	4	2
Машина В	5	1	3	4	7	4

Для выполнения Технического Задания следует использовать средства автоматизированной обработки расписаний, которые обеспечивают программную реализацию процедур синтеза и анализа расписаний на основе формального аппарата теории расписаний.

## Автоматизация составления расписаний

Для автоматизации процедур составления оптимальных расписаний в условиях задачи 2-х машин может быть применен программно-методический комплекс SHD2. Программный комплекс SHD2 разработан для эксплуатации в среде OS UNIX. Он поддерживает прозрачный режим алфавитно-цифрового ввода-вывода на терминалах, совместимых с VT100 или эмулирующих его характеристики. В состав комплекса SHD2 входит выполняемый модуль shd2, ресурсный текстовый файл shd2.rs и стартовый командный файл shd2.csh, который удобно применять для запуска комплекса в оконно-графической среде X Window System. Эта среда de-facto входит в комплект поставки современных версий OS UNIX.

Все компоненты комплекса SHD2 должны быть расположены в одном каталоге файловой системы OS UNIX, например, \$HOME/shd2, который целесообразно сделать текущим перед запуском комплекса, используя, например, команду:

```
% cd $HOME/shd2 .
```

Запуск комплекса SHD2 можно тогда осуществить по базовому имени выполняемого модуля (shd2). В любом случае комплекс SHD2 можно запускать по полному маршрутному имени выполняемого модуля, например,

```
% $HOME/shd2/shd2,
```

вне зависимости от выбора текущего каталога. В среде X Window System комплекс SHD2 может работать в окнах клиентной программы эмулятора алфавитно-цифрового терминала xterm, где его запуск осуществляется в порядке, аналогичном рассмотренному выше. Одновременную активизацию эмулятора терминала xterm и комплекса SHD2 с дополнительной сервисной настройкой оконных ресурсов обеспечивает стартовый командный файл shd2.csh, который можно запустить из любого активного окна эмулятора терминала (желательно в фоновом режиме - &) с помощью интерпретатора команд /bin/csh следующим образом:

```
% /bin/csh $HOME/shd2/shd2.csh & ,
```

или с помощью интерактивных средств оконного менеджера X Window System, например, рабочей области Workspace оконного менеджера olwm (Open Look Window Manager), при соответствующей настройке его ресурсных файлов, в частности, файла \$HOME/.openwin-menu, где должна быть указана метка вызова стартового файла комплекса SHD2, например, "Schedule CAD".

Вне зависимости от способа запуска комплекс SHD2 обладает следующими функциональными возможностями:

- инициализация и редактирование состава партии и длительностей операций работ на машинах А и В;
- формирование расписаний на основе экспертных и эвристических соображений;
- оценка оптимальности текущего варианта расписания по правилу Джонсона;
- поиск оптимальных расписаний с помощью перестановочных приемов, реализующих алгоритмы Джонсона и приоритетов;
- модификация и вариация оптимальных расписаний, удовлетворяющих правилу Джонсона;
- перечисление расписаний путем полного перебора перестановок работ партии;
- графическая иллюстрация расписаний в форме диаграммы Ганта;
- имитация процесса обслуживания по текущему расписанию в режиме модельного времени;

- дифференцированный автоконтроль практических навыков составления оптимальных расписаний.

Интерактивную поддержку функциональных возможностей комплекса SHD2 обеспечивает система иерархически взаимосвязанных экранных сцен. Их форматы заданы в ресурсном файле shd2.rs. Каждая сцена снабжена экранной инструкцией по обработке. Управление обработкой и иерархией сцен осуществляется в форме пассивного диалога, который реализуется через управляющие меню. Эти меню обеспечивают выбор требуемых сцен и управление их обработкой с помощью алфавитно-цифровой клавиатуры. Последовательность сцен, чередуемых в одном сеансе работы с комплексом SHD2, образует сценарий диалога. Основу любого осмысленного сценария диалога образуют 7 экранных сцен, краткое описание которых приведено ниже.

Сцена Main Menu предоставляет управляющее меню верхнего уровня. Это меню образуют 2 колонки пунктов. В левой колонке сосредоточены инструментальные сцены упорядочивания работ партии, в правой - сервисные сцены интерпретации результатов упорядочивания. Из основного меню доступны сервисные сцены (Service Tools):

- **Job Party Time Editor,**
- **Jobs Schedule Gantt Chart,**
- **Imit Job Party Processing**

и инструментальные сцены (Job Order):

- **Optimal Job Order,**
- **Cut-&-Try Job Order,**
- **Examine Job Order.**

Сцена Job Party Editor используется для редактирования состава партии и длительностей операций работ. В состав партии может быть включено до 26 работ, обозначаемых строчными буквами латинского алфавита. Длительности операций работ могут быть назначены в пределах от 0 до 99 единиц. После сохранения результатов редактирования работы партии будут упорядочены по алфавиту. Эта сцена используется для подготовки исходных данных в соответствии с техническим заданием.

Сцена Jobs Schedule Gantt Chart используется для визуализации графика расписания процесса обслуживания текущей перестановки работ по диаграмме Ганта. Сцена позволяет определить загрузку машин А и В для каждого момента времени от начала до конца процесса обслуживания. Она отражает численные характеристики расписания: суммарная длительность простоя и функционирования каждой машины, а также общую продолжительность процесса обслуживания.

Сцена *Imit Job Party Processing* имитирует выполнение процесса обслуживания по текущему расписанию в пошаговом и автоматическом режиме. Она отображает динамику прохождения работ по машинам и состояние очереди работ между машинами в каждый момент модельного времени. Эта сцена может быть использована для оценки емкости буфера хранения работ между машинами при реализации процесса обслуживания.

Сцена *Optimal Job Order* предоставляет управляющее меню второго уровня, которое используется для выполнения и иллюстрации перестановочных приемов оптимального упорядочивания работ партии. В состав этого меню входят функциональные и сервисные сцены. Функциональные сцены реализуют алгоритмы Джонсона и приоритетов, а также интерактивное перечисление класса оптимальных перестановок работ, удовлетворяющих правилу Джонсона. Сервисные сцены иллюстрируют перестановочные приемы и правило Джонсона.

Сцена *Cut-&-Try Job Order* применяется для перечисления перестановок работ методом полного перебора. Сцена позволяет реализовать перебор в пошаговом и автоматическом режиме. Автоматический перебор будет всегда приостановлен, когда достигнута перестановка работ, для которой длительность расписания процесса обслуживания не больше, чем у любой из перечисленных ранее. Перебор перестановок можно принудительно прервать на любой итерации с сохранением текущей достигнутой перестановки или лучшей по длительности перестановки из числа перечисленных ранее. Сцена *Examine Job Order* используется для интерактивного контроля практических навыков составления оптимальных расписаний с помощью перестановочных приемов или правила Джонсона. На время выполнения этой контрольной сцены длительности операций работ партии меняются случайным образом. Результаты контроля оцениваются по 100-бальной шкале и отображаются штрафным табло сцены. Для выбора типа контроля в сцене используется управляющее меню, пункты которого вызывают соответствующие сцены контроля алгоритма Джонсона, алгоритма приоритетов и правила Джонсона. Идеальным результатом контроля является нулевой штрафной балл по всем 3-м перечисленным контрольным сценам.

## **Порядок выполнения технического задания**

1. В сеансе многопользовательского режима OS UNIX запустить комплекс SHD2. При успехе запуска на экране отображается входная заставка комплекса, содержащее его полное название. Нажать клавишу <Enter> или <Return> для вызова сцены основного меню комплекса SHD2.

2. Ввести исходные данные в соответствии с Техническим Заданием, установив состав партии и длительности операций работ средствами сцены *Job Party Editor* основного меню. После фиксации исходных данных будет сформирован исходный вариант расписания простого процесса обслуживания, где работы упорядочены по алфавиту в соответствии с их обозначениями.



3. Построить диаграмму Ганта для исходного варианта расписания, используя сцену Jobs Schedule Gantt Chart основного меню. Зафиксировать суммарную длительность простоя машины В и общую продолжительность процесса обслуживания по исходному варианту расписания.
4. Оценить качество исходного расписания по правилу Джонсона, используя сцену Show Johnson Rule Optimal Test управляющего меню Optimal Job Order. Отметить, для каких пар соседних работ нарушено условие Джонсона.
5. Найти оптимальную перестановку работ партии с помощью алгоритма Джонсона, используя сцену Johnson Job Order управляющего меню Optimal Job Order.
6. Проанализировать этапы преобразования исходной перестановки работ по шагам алгоритма Джонсона, используя сцену Johnson Order Trick управляющего меню Optimal Job Order.
7. Перечислить все возможные варианты оптимальных перестановок работ путем транспозиции соседних работ с равными продолжительностями коротких операций, используя сцену Modify Johnson Order из управляющего меню Optimal Job Order.
8. Найти оптимальную перестановку работ с помощью алгоритма приоритетов, используя сцену Priority Job Order из управляющего меню Optimal Job Order.
9. Зафиксировать приоритеты работ, назначенные алгоритмом приоритетов, используя сцену Priority Order Trick управляющего меню Optimal Job Order.
10. Перечислить все возможные варианты оптимальных перестановок работ путем транспозиции работ с равными приоритетами, используя сцену Variate Priority Order управляющего меню Optimal Job Order.
11. Сопоставить классы оптимальных расписаний, которые получены алгоритмами Джонсона и приоритетов. Зафиксировать общее число оптимальных расписаний, удовлетворяющих правилу Джонсона.
12. Построить диаграмму Ганта для текущего варианта оптимального расписания, используя сцену Jobs Schedule Gantt Chart основного меню. Зафиксировать суммарную длительность простоя машины В и общую продолжительность процесса обслуживания по оптимальному расписанию.
13. Имитировать выполнение процесса обслуживания по найденному варианту оптимального расписания в режиме модельного времени, используя сцену Imit Job Party Processing из основного меню. Оценить максимальную длину очереди и требуемую емкость буфера накопления работ (pend buffer) между машинами А и В в процессе обслуживания.

14. Найти методом полного перебора произвольное расписание, равное оптимальному, по длительности процесса обслуживания, перестановка работ которого не удовлетворяет правилу Джонсона, используя сцену Cut-&-Try Job Order. Зафиксировать полученную методом полного перебора перестановку работ партии и отметить любую пару соседних работ, для которых нарушено правило Джонсона.

15. Определить методом полного перебора общее число оптимальных расписаний, используя сцену Cut-&-Try Job Order основного меню.

16. Составить отчет по результатам выполнения технического задания. Отчет должен содержать: исходные данные, список оптимальных перестановок работ партии, полученных с помощью перестановочных приемов и метода полного перебора, диаграммы Ганта для исходного и любого оптимального расписания процесса обслуживания, ресурсные характеристики оптимального расписания.

## **Организация самоконтроля**

Функциональные возможности комплекса SHD2 дополняют средства автоматизации самоконтроля практических навыков составления оптимальных расписаний, реализованные экранной сценой Examine Job Order основного меню. Эта сцена обеспечивает интерактивную поддержку самоконтроля по 3-м пунктам, в которых требуется составить оптимальное расписание процесса обслуживания заданной партии работ, применяя алгоритм Джонсона, алгоритм приоритетов и правило Джонсона, соответственно.

При активизации сцены Examine Job Order по каждому пункту контроля назначается штрафной балл, равный 99. Выполнив последовательно пункты контроля, перечисленные выше, нужно уменьшить штрафные баллы до 0. Текущие результаты контроля отражает штрафное табло сцены. Для выбора пункта контроля используется управляющее меню. При каждом выборе любого контрольного пункта длительности операций работ изменяются случайным образом. Если при выполнении выбранного пункта контроля не удалось получить оптимальную перестановку работ контрольной партии, вызывается соответствующая обучающая сцена, которая демонстрирует, как достигается желаемый оптимальный результат.

Для самоконтроля по алгоритму Джонсона нужно из управляющего меню сцены Examine Job Order выбрать контрольную сцену Examine Johnson Trick. В этой сцене необходимо переместить (transfer) работы предлагаемой исходной (source) перестановки на свободные места формируемой (target) оптимальной перестановки в соответствии с алгоритмом Джонсона, используя экранное табло длительностей операций. Число ошибок при размещении работ в оптимальной перестановке определяет штрафной балл по этому пункту контроля.

Для самоконтроля по алгоритму приоритетов используется контрольная сцена Examine Priority Trick управляющего меню сцены Examine Job Order. В этой сцене

необходимо построить оптимальную перестановку работ контрольной партии, назначая (assign) приоритеты работ и сортируя (sort) работы в соответствии с алгоритмом приоритетов. Для назначения приоритетов предоставляется экранное табло длительностей операций работ. Штрафной балл по этому пункту контроля равен числу ошибок назначения приоритетов работ.

Для самоконтроля по правилу Джонсона нужно из управляющего меню сцены Examine Job Order выбрать контрольную сцену Examine Johnson Rule. В этой сцене необходимо получить оптимальную перестановку работ путем транспозиции (swap) соседних работ, для которых нарушено условие Джонсона по экранному табло соотношений длительностей операций. Сцена позволяет интерактивно оценить (estimate) корректность транспозиций, сообщая число нарушений условия Джонсона для текущей перестановки работ. Штрафной балл по данному пункту контроля равен числу оценок транспозиций, использованных при построении оптимальной перестановки работ контрольной партии.

## **СПИСОК ЛИТЕРАТУРЫ**

1. Р.В. Конвей, В.Л. Максвелл, Л.В. Миллер Теория расписаний. М., Наука, 1975 г.
2. В.С. Танаев, В.В. Шкурба Введение в теорию расписаний. М., Наука, 1975 г.
3. А. Кофман Введение в прикладную комбинаторику. М., Наука, 1975 г.
4. А.А. Корбут, Ю.Ю. Финкельштейн Дискретное программирование. М., Наука, 1969 г.