Алгоритм Джонсона.

Рассмотрим следующую постановку.

Имеется 5 процессов А1, А2,..,А5 и 2 ресурса (процессоры) Р1, Р2.Каждый процесс выполняется на ресурсе Р1, потом на Р2. Известно время выполнения каждого процесса на каждом ресурсе.

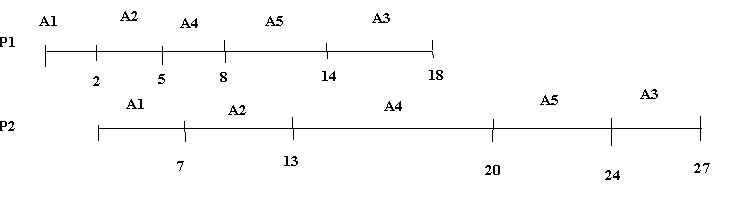
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | А1 | А2 | А3 | А4 | А5 |
| Р1 | 2 | 3 | 4 | 3 | 6 |
| Р2 | 5 | 6 | 3 | 7 | 4 |

Надо построить расписание (порядок выполнения работ), при кот. время завершения будет наименьшим.

Находим наим. значение в матрице. Если оно в первой строке, то помещаем процесс в начало расписания , если во второй - то в конец.

Здесь наим. значение – 2. Значит помещаем А1 в начало расписания. След. Пойдет процесс А2, А4.

В конец расписания (на второй ресурс помещаем А3, затем А5, а потом они переходят на первый ресурс). Получаем след.картину:



Мин.время выполнеия 27 ед. на обоих ресурсах.

Алгоритм банкира (АБ).

АБ – способ использования общего ресурса при его малом количестве. Раньше использовался для памяти. Пример .

Объем операт.памяти = 100Мб.

A,B,C – процессы. Для каждого процесса известен max Vпамяти, кот.может ему потребоваться.

Пусть имеем такие запросы процессов на память.

|  |  |  |
| --- | --- | --- |
|  | запрос | Максимально для завершения |
| A | 20 | 40 |
| B | 40 | 45 |
| C | 30 | 36 |

Выделяем память процессу, если он гарантированно завершиться.

Рассм. Процесс А. Ему для работы надо 20 Мб, но чтобы завершиться – 40 Мб. У нас 100Мб. 20+40=60Мб. У нас хватает памяти, тогда выделяем ее процессу А**, НО выделяем только 20** (по запросу, а не максимально!).

Получаем 100-20= 80Мб у нас осталось.

Смотрим след.процесс B. В данный момент ему надо 40, а максимально 40+45=85Мб. У нас только в наличии 80, поэтому ему ничего не выделяем.

Смотрим процесс С. Ему в данный момент надо 30, а всего для завершения – 66Мб. У нас те же 80Мб. Значит для процесса С выделяем память, но выделяем только 30Мб!

Банкир дает только в том случае, если памяти хватит по максимуму последнему процессу. Остаток можно распределить для следующих процессов. Когда память из-под процессов А или С освободиться, то можно выделить для процесса В, если ее хватает.