## 3-2 最优批处理问题

### 问题描述:

在一台超级计算机上，编号为1，2……n的n个作业等待批处理。批处理的任务就是将这n个作业分成若干批，每批包含相邻的若干作业。从时刻0开始，分批加工这些作业。在每批作业开始前，机器需要启动时间s,而完成这批作业所需的时间是单独完成批中各个作业需要时间的总和。单独完成第i个作业所需的时间是ti，所需的费用是它的完成时刻乘以一个费用系数fi。同一批作业将在同一时刻完成。例如，如果在时刻T开始一批作业x,x+1,,x+k,则这一批作业的完成时刻均为T+S+(tx, +tx+1 +...+tx+k)。最优批处理问题就是要确定总费用最小的批处理方案。

例如，假定有5个作业等待批处理，且S = 1, (t1,t2,t3,t4,t5)=(1,3,4,2,1), (f1,f2,f3,f4,f5)=(3,2,3,3,4)如果采用批处理方案{1,2},{3},{4,5}，则各作业的完成时间分别为(5,5,10,14,14)，各作业的费用分别为(15,10,30,42,56)，因此，这个批处理方案总费用是153。

### 算法设计:

对于给定的待批处理的n个作业，计算其总费用最小的批处理方案。

### 数据输入：

由文件input.txt提供输入数据。文件的第1行是待批处理的的作业数n, 第2行是启动时间S。接下来每行有2个数，分别为单独完成第i个作业所需要的时间是ti和所需要的费用系数fi。

### 结果输出：

将计算出的最小总费用输出到文件output.txt中。

### 最优子结构

若{1,2,...i}{i+1,i+2....}.....{j,j+1,...}...{...,n-2,n-1,n}是最优解，则易知不考虑前j-1个时，{j,j+1,...}...{...,n-2,n-1,n}是后j到n个的最优解。否则存在更优解使得{1,2,...i}{i+1,i+2....}.....{j,j+1,...}...{...,n-2,n-1,n}不是最优解。

因此具有最优子结构。

### 递归计算最优值

设cost[i]是第i个作业到第n个作业的最小总费用减去前i-1个作业的时间乘第i个作业到第n个作业的费用和。则有：

### 构造最优解

### 计算复杂性

5

1

1 3

3 2

4 3

2 3

1 4

## 4

### 最优子结构

### 递归计算最优值

### 构造最优解

### 计算复杂性

## 6

### 最优子结构

### 递归计算最优值

### 构造最优解

### 计算复杂性

## 7

### 最优子结构

### 递归计算最优值

### 构造最优解

### 计算复杂性

## 13

### 最优子结构

### 递归计算最优值

### 构造最优解

### 计算复杂性