### 一、问题介绍与要求

#### 题目

**资源分配问题：**某厂根据计划安排，拟将*m*台相同的设备分配给*n*个车间，各车间获得这种设备后，可以为国家提供盈利Cij(分配*i*台设备*j*号车间将得到的利润，1≤*i*≤*m*，1≤*j*≤*n*) 。问如何分配，才能得到最大的盈利？

#### 题目问题的本质

将数量一定的一种或若干种资源合理分配给若干个使用者或地区，使得目标函数最优。

#### 要求

* 设计动态规划算法求解资源分配问题，写出求得最优值的递推公式。
* 测试不同问题规模（按级数增长）的执行结果（即最优分配方案、最优分配方案的值）、运行时间。
* 能够实现的问题规模越大，成绩越高。

### 二、解决思路

#### 动态规划算法

动态规划是用来解决多阶段决策过程最优化的一种数量方法。其特点在于，**它可以把一个n维决策问题变换为几个一维最优化问题**，从而一个一个地去解决。

#### 资源分配问题与动态规划

资源分配问题要求把资源分配给**多个使用者**，本质上来说就是一个**多维（分阶段）决策问题**，系统所处的状态和时刻（即某个时候剩下的资源）是进行决策的重要因素，我们可以运用动态规划来解决分配问题。

此外，资源分配问题满足：

* 最优子结构性质：如果你分配给前n的使用者的方案是最优的，那么就能找到分配给前n+1个使用者的最优方案。
* 无后效性：分配给前n个使用者的最优方案并不会受到后续使用者的影响。
* 子问题重叠性质：每次求下一阶段的最优解的时候总是要访问前几个阶段的情况，因此由很多重叠子问题，可以填表来解决问题。

#### 具体算法

设：

* m为设备总量，总共有n个车间。按车间划分阶段，k=1，2，…，n；
* 决策变量fork，表示分配给第k个使用者的资源数量；
* 状态变量为，residuek表示在第k阶段，剩余的可以分给第k个至第n个使用者的总资源数量；
* 状态转移方程：，其中
* 允许决策集合：
* 阶段指标函数(即Gij表)：

表示分配给第k个车间fork设备时的收益；

则动态规划转移方程为：



代码化：

* 我们用状态量表示用i台设备分配给前j个车间的最大获利，那么显然有：
* ，其中0<=k<=i。
* 即最大利润分为两部分，一部分是分配k台给前j-1个车间，分配i-k给第j个车间
* 依次填表，我们可以得到整张表(程序实现时先枚举车间数，再枚举设备数，再枚举状态转移时用到的设备数，简单3重for循环语句即可完成)。
* 再用l[i][j]表示获得最优解时第j号车间使用的设备数为i-l[i][j]，于是从结果倒推往回求即可得到分配方案。

#### 性能分析

* 时间复杂度为，空间复杂度为，倘若此题只需求最大获利而不必求方案，则状态量可以减少一维，空间复杂度优化为。
* 题目要求尽量做大问题规模，即优化到以下，但是经过思考，由于本次实验使用的是JAVA语言，对地址空间控制力不足，故无法做大幅度优化。如需优化必须替换为C语言，利用指针来优化，循环使用数组空间。

### 三、算法的对应代码展示

#### 随机生成利润表算法

##### 代码

##### 解释

#### 初始化表

##### 代码

##### 解释

#### 填充动态规划状态表f

##### 代码

##### 解释

#### 查表求结果

##### 代码

##### 解释

### 四、算法举例

假设利润表如下：

则生成的f表如下：

生成的l表如下：

解决方案查表过程：