**算法设计与分析**

**实验报告**

|  |  |
| --- | --- |
| **专业班级** | **计算机科学与技术21-1班** |
| **学生姓名** | **何超** |
| **学 号** | **202001021107** |
| **指导老师** | **张鹏** |

**山东科技大学**

|  |  |  |  |
| --- | --- | --- | --- |
| **实验名称** | 独立任务最优调度问题 | | |
| **实验日期** | **2022/06/10** | **实验地点** | J13-332 |
| **指导老师** | **张鹏** | **实验成绩** |  |

1. **实验目的和要求：**

**实验目的：**

掌握[动态规划](https://so.csdn.net/so/search?q=%E5%8A%A8%E6%80%81%E8%A7%84%E5%88%92&spm=1001.2101.3001.7020)的基本思想和解决问题的基本步骤，认识不同题目中动态规划递推关系式的设计。

**实验要求：**

用c++语言实现用动态规划算法解决独立任务最优调度问题，体会动态规划算法解决问题的基本思路和步骤。

1. **实验内容：**

**解决问题：**

用2台处理机A和B处理n个作业。设第i个作业交给机器A处理时需要时间ai，若由机器B来处理，则需要时间bi。由于各作业的特点和机器的性能关系，很可能对于某些i，有ai>=bi，而对于某些j,j≠i，有aj<bj。既不能将一个作业分开由2台机器处理，也没有一台机器能同时处理2个作业。设计一个动态规划算法，使得这2台机器处理完这n个作业的时间最短(从任何一台机器开工到最后一台机器停工的总时间)。

**具体内容：**

由题目可知：每个任务仅有两个选择：由A处理，或者由B处理。解题的目标是合理安排每个任务由哪个机器处理，从而找到最短的处理完所有任务的时间。所有任务都是需要被处理的。那么每个任务的处理顺序是不重要的。所以，可以从第一个任务开始，按顺序决定每个任务由哪个机器处理，直到处理完所有的任务。

1. **算法描述及实验步骤：**

**总体设计思路：**

设布尔量p[i][j][k]表示前k个作业是否可以在处理机A用时不超过i且处理机B用时不超过j时间内完成。由上述分析，我们可以知道：当i>=sum\_a，或者j>=sum\_b时，p[i][j][k]一定为真。那么我们可以使用穷举法，将机器A、B花费的时间分别从0枚举至sum\_a和sum\_b，对于每种情况，都求出p的bool值，再从中选择出最短时间即可。p[i][j][k]是一个bool值，其真假表示a机器花费i时间、b机器花费j时间的前提下，是否可以完成k个任务。我们这里假设任务都是按顺序处理。也就是说，假设机器A花费时间i、机器B花费时间j，若可以在此时间内完成至任务k，那么p[i][j][k]=true，否则为false。当求得所有p的值后，找到所有值为true的p，选出min(max(i,j))就是问题的解。取max是因为最终结果取决于A和B中花费时间较长的那一个。再取min寻找出所有能够完成任务的(i,j)组合中的最小值。

**递推关系：**

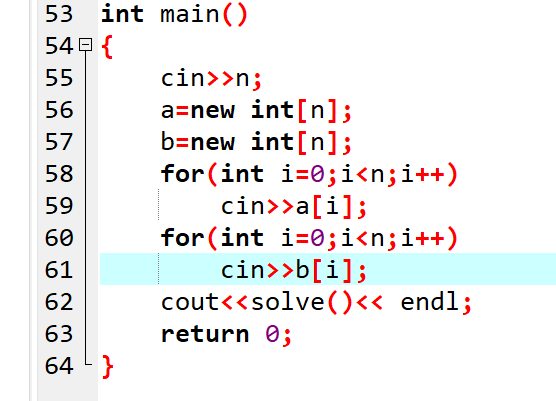
求出p[i][j][k]的主要过程主要是求出动态规划过程的递推关系，第k个任务和第k-1个任务有密不可分的关系。若p[0..sum\_a][0..sum\_b][0..k-1]已经全部求解完毕。那么完成第k件任务有两个选择：分配给A机器，或者B机器。假设A花费时间为i、B花费时间为j，可以完成任务k-1，则A花费时间ak+i、B花费时间j，一定可以完成任务k。若A花费i时间、B花费bk+j时间，也可以完成任务k。因为这相当于将k分别交给A或者B完成。将该逻辑关系写成以p表示的形式，就是所求的递推关系式：p[i][j][k]=p[i-ak][j][k-1] || p[i][j-bk][k-1]

1. **实验调试过程及输入输出结果：**

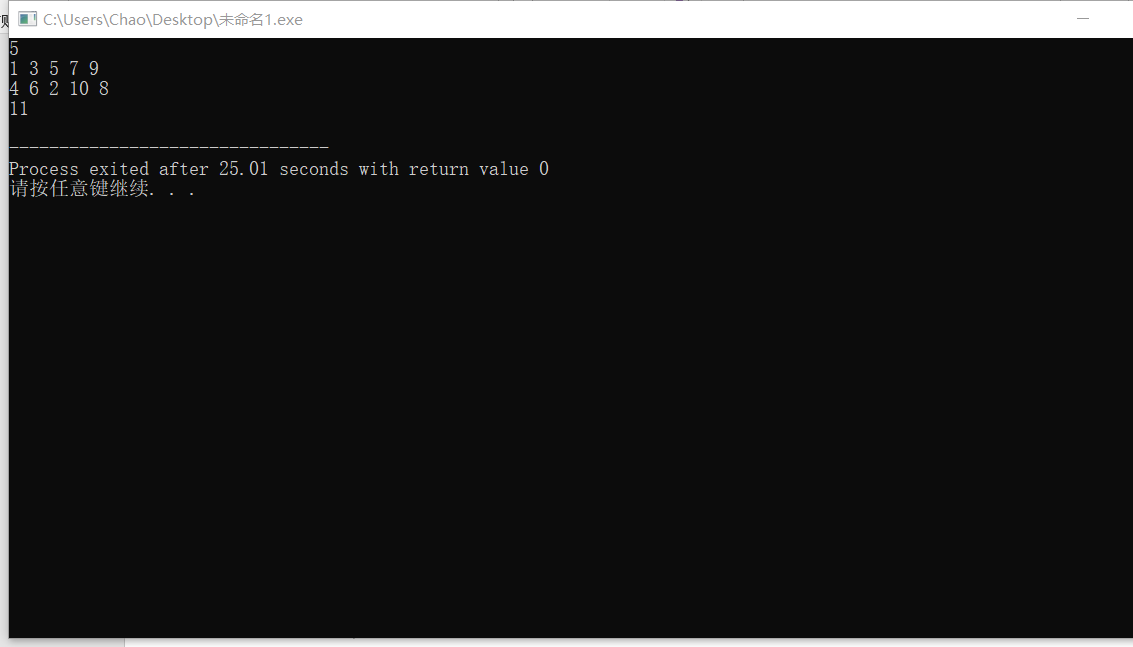
根据上述设计的思路与动态规划思想，在确定了递推公式p[i][j][k]=p[i-ak][j][k-1] || p[i][j-bk][k-1]后，编写函数solve（）实现该过程，首先建立三维数组p[][][],而后具体实现如下：



编写函数main，向数组中输入a,b机器处理作业耗时，后调用solve函数打印结果，编写代码如下：



编写测试数据：任务数量5，a机器耗时1 3 5 7 9，b机器耗时4 6 2 10 8，运行结果正确，如下：



**五、实验总结：**

在本次实验中，我们使用动态规划思想设计解决了独立任务最优调度问题，本题目即求解固定数量的任务在有限的机器资源下求解最快完成时间的问题。根据问题的这种特性我们决定利用动态规划思想解决。在分析问题的过程中，重要的是了解问题的逻辑关系。在编写代码的过程中，要注意代码编写的良好风格，以及内存申请的方法。经过本次实验，我们对动态规划分析问题，寻找递推关系式的理解进一步加深了。