

2020 年 8 月 9 日

1.P2722 [USACO3.1]总分 Score Inflation

算法思路：

因为是连着做（8月8日的题），所以还是很明显的看出来是背包问题，主要是时间限制和分数，可以很明显的对应“重量”和“价格”。一开始没有仔细读题，以为是01背包，然后测例的输出是505，对比答案的605，在想怎么会是完全背包咧？然后再读一遍题目，原来每一行的时间花费对应的是一类题目，我还以为是一道题目咧~

代码：

```
#include <iostream>
using namespace std;
#define N 10010
int dp[N] = {0}, w[N], v[N];
int main(){
    int m, n;
    cin >> m >> n;
    for(int i = 0; i < n; i++){
        cin >> v[i] >> w[i];
    }
    for(int i = 0; i < n; i++){
        for(int j = w[i]; j <= m; j++){
            dp[j] = max(dp[j], dp[j-w[i]]+v[i]);
        }
    }
    cout << dp[m];
    return 0;
}
```

Accepted截图：



wongsiyoung

所属题目 P2722 [USACO3.1]总分 Score Inflation

评测状态 Accepted

评测分数 100

提交时间 2020-08-09 14:25:33

2.[P1387 最大正方形]]

(<https://www.luogu.com.cn/problem/P1387>)

算法思路:

以为还是用背包的思维去做就不会了~, 然后就看了姐姐的题解

min是因为如果有一个是0那就是0拉, 要大家是1才可以继续围成正方形, 所以看左上方, 左边, 上边的最小值是否为1~

代码:

```
#include <iostream>
#include <cstring>
using namespace std;

#define N 110
int dp[N][N];

int main(){
    int n,m;
    cin >> n >> m;
    for(int i = 0; i < n; i++){
        for(int j = 0; j < m; j++){
            cin >> dp[i][j];
        }
    }
    int res = 0;
    for(int i = 1; i < n; i++){
        for(int j = 1; j < m; j++){
            if(dp[i][j] == 1){
                dp[i][j] = min(dp[i-1][j-1], min(dp[i-1][j], dp[i][j-1]))+1;
            }
            res = max(res, dp[i][j]);
        }
    }
    //int res = dp[0][0];
    /*for(int i = 0; i < n; i++){
```

```
        for(int j = 0; j < m; j++){
            res = max(res,dp[i][j]);
        }
    }*/
    cout << res;
    return 0;
}
```

Accepted截图：

 wongsiyoung

所属题目	P1387 最大正方形
评测状态	Accepted
评测分数	100
提交时间	2020-08-09 14:45:25