

股票市场

一. 考察内容:

动态规划

二. 题目分析:

[题目大意]

给出每天的股票的价格，你可以用你手中的钱自由买卖，每个股票只能买整数个，求出整体最大获利。

[写题思路]

首先，我们考虑一个股票第一天买进，并在第三天卖出，情况很复杂，我们可以把他化为第一天买进，并在第二天卖出，第二天再次买进，又在第三天卖出的总获利，所以，我们每天都可以把前一天的股票都换成钱，再重新选择需要买的股票。

这时，问题变为每天，有k元钱，和n个股票，每个股票可以买无限个，每个股票买一次可以赚a_i元，是一道典型的完全背包问题，我们只需要对于每一天做一次完全背包即可。

三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATE
/*****
*创建时间: 2018 09 06
*文件类型: 源代码文件
*题目来源: BZOJ
*当前状态: 已通过
*备忘录: 贪心 动态规划
*作者: HtBest
*****/
#include <stdio.h>
#include <iostream>
#include <string>
#include <stdlib.h>
#include <string.h>
#include <algorithm>
#include <queue>
#include <bitset>
// #include <sys/wait.h>
// #include <sys/types.h>
// #include <unistd.h>
using namespace std;
int n,m,K,day,f[500001],a[51][11];
/* Variable explain:

*/
void read()
{
    scanf("%d%d%d",&n,&m,&K);
    for(int i=1;i<=n;++i)
    {
        for(int j=1;j<=m;++j)scanf("%d",&a[i][j]);
    }
    return;
}
void dp()
{
    for(int i=2;i<=m;++i)
    {
        for(int j=1;j<=K;++j)f[j]=0;
        for(int j=1;j<=n;++j)
```

```
        {
            for(int k=a[j][i-1];k<=K;++k)
            {
                f[k]=max(f[k],f[k-a[j][i-1]]+a[j][i]-a[j][i-1]);
            }
            K+=f[K];
        }
        printf("%d\n",K);
    }
    int main()
    {
        // freopen(".in","r",stdin);
        // freopen(".out","w",stdout);
        read();
        dp();
        return 0;
    }
```

[<题目跳转>](#) [<查看代码>](#)