## 2566 统计硬币

### 一、题目

#### 问题描述

假设一堆由1分、2分、5分组成的n个硬币总面值为m分，求一共有多少种可能的组合方式（某种面值的硬币可以数量可以为0）。

#### 输入数据

输入数据第一行有一个正整数T，表示有T组测试数据； 接下来的T行，每行有两个数n,m，n和m的含义同上。

#### 输出数据

对于每组测试数据，请输出可能的组合方式数； 每组输出占一行。

#### 输入样例

2 3 5 4 8

#### 输出样例

1 2

#### 题目来源

HDU 2566 http://acm.hdu.edu.cn/showproblem.php?pid=2566

### 二、题解

#### 解题思路

这题可以通过暴力穷举法。

当然，这也是一道完全背包问题。我们可以将其化为01背包，只不过每种物品可以重复取的各数和为n就行。

#### 参考程序

一 完全背包

#include<stdio.h>  
#include<string.h>  
int dp[1002][1002]; //如果放在主函数里会造成栈溢出  
//dp[j][k]存用j个硬币，组成k分钱的方案数  
int main()  
{  
 int t;  
 int a[3]= {1,2,5};  
 scanf("%d",&t);  
 while(t--)  
 {  
 int n,m;  
 scanf("%d%d",&n,&m);  
 memset(dp,0,sizeof(dp)); // 给字符数组赋值时可以赋任意值，  
 //但是给int型的数组赋值时要注意，一般只赋值为-1， 0， 127  
 //在#include<string.h>里  
 dp[0][0]=1;  
 for(int i=0; i<3; i++)  
 for(int j=1; j<=n; j++)  
 for(int k=a[i]; k<=m; k++)  
 dp[j][k]+=dp[j-1][k-a[i]];  
 printf("%d\n",dp[n][m]);  
 }  
  
 return 0;  
}

二 暴力穷举

#include<stdio.h>  
int main()  
{  
 int t,n,m,i,j,k,ans;  
 scanf("%d",&t);  
 while(t--)  
 {  
 ans=0;  
 scanf("%d%d",&n,&m);  
 for(i=0;i<=m;i++)  
 for(j=0;j<=m/2;j++)  
 for(k=0;k<=m/5;k++)  
 if(i+2\*j+5\*k==m&&i+j+k==n)  
 ans++;  
 printf("%d\n",ans);  
 }  
 return 0;  
}

#### 复杂度分析

无

#### 编程技巧

无