

背包专题之混合背包

混合背包是指在给定的物品中，有的可以装无限个，有的则只能装入有限个。那么解题的思路就很明显了，即把所有的物品按照完全背包和多重背包两类进行分类，然后求解即可。

那么具体该如何分类呢？这里有一个技巧，当我们尝试向背包装入物品时，如果当前物品可放置的数量大于背包的容量，那么对于背包来说，这个物品就是一个无限量装入的物品，那么对其就可以使用完全背包的解法。

否则，我们就按照多重的背包进行求解。

1270：混合背包

题目描述

一个旅行者有一个最多能装 V 公斤的背包，现在有 n 件物品，它们的重量分别是 W_1, W_2, \dots, W_n ，它们的价值分别为 C_1, C_2, \dots, C_n 。有的物品只可以取一次（01背包），有的物品可以取无限次（完全背包），有的物品可以取的次数有一个上限（多重背包）。求解将哪些物品装入背包可使这些物品的费用总和不超过背包容量，且价值总和最大。

输入

第一行：二个整数， M （背包容量， $M \leq 200$ ）， N （物品数量， $N \leq 30$ ）；

第 $2..N+1$ 行：每行三个整数 W_i, C_i, P_i ，前两个整数分别表示每个物品的重量，价值，第三个整数若为0，则说明此物品可以购买无数件，若为其他数字，则为此物品可购买的最多件数(P_i)。

输出

仅一行，一个数，表示最大总价值。

输入样例

```
10 3
2 1 0
3 3 1
4 5 4
```

输出样例

```
11
```

编码

```
#include<bits/stdc++.h>
```

```

using namespace std;

int bagV, n;

int w[31];           //商品的体积
int v[31];           //商品的价值
int k[31];           //商品的数量

int dp[201];         //动态规划表

int main() {
    //记录最大承重和物品数量
    cin >> bagV >> n;
    //记录每个物品的重量和价值
    for (int i = 1; i <= n; i++) {
        cin >> w[i] >> v[i] >> k[i];
    }
    //从放入第一件物品开始
    for (int i = 1; i <= n; i++) {
        //物品可放置的数量大于了背包容量，直接按照完全背包解法执行
        if (k[i] * w[i] >= bagV || k[i] == 0) {
            for (int j = w[i]; j <= bagV; j++) {
                dp[j] = max(dp[j], dp[j - w[i]] + v[i]);
            }
        }
        //二进制分解，按照01背包进行处理
        //依次尝试k的数量为1,2,4,8，将其每一个分组当成独立的物品进行判断
        else {
            int tk = k[i];
            int ck = 1;
            //减少循环数量
            while (ck <= tk) {
                //此处按照01背包的一维优化进行装包处理
                for (int j = bagV; j >= ck * w[i]; j--) {
                    //避免建立更多的数组
                    dp[j] = max(dp[j], dp[j - ck * w[i]] + ck * v[i]);
                }
                tk -= ck;
                ck <<= 1;
            }
            //处理没有用尽的数量
            if (tk) {
                for (int j = bagV; j >= tk * w[i]; j--) {
                    dp[j] = max(dp[j], dp[j - tk * w[i]] + tk * v[i]);
                }
            }
        }
    }
}

```

```
    //最大值在最后一个格子中  
    cout << dp[bagV];  
    return 0;  
}
```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。



);