

# 01背包练习

## 人员

于家瑞、洪晨棋、洪晨栋、于霄龙、陶汇笙、郭栩睿 到课, 王恩泽、崔宸赫 线上

## 上周作业检查

上周作业链接: <http://cppoj.kids123code.com/contest/1784>

		比赛概况	题目列表	选择题列表	提交记录	★ 实时榜单	★ 选择题排行榜				
#	用户名	姓名	编程分	时间	A	B	C	D	E	F	
1	yuxiaolong	于霄龙	600	233	100	100	100	100	100	100	
2	hongchenqi	洪晨棋	600	236	100	100	100	100	100	100	
3	yujiarui	于家瑞	600	237	100	100	100	100	100	100	
4	hongchendong	洪晨栋	600	241	100	100	100	100	100	100	
5	guoxurui	郭栩睿	600	256	100	100	100	100	100	100	
6	taohuisheng	陶汇笙	600	258	100	100	100	100	100	100	
7	cuchenhe	崔宸赫	300	141	100	100	100				
8	wangenze	王恩泽	100	17	100						

## 本周作业

<https://cppoj.kids123code.com/contest/1906> (课上讲了 A ~ D 这些题, 课后作业是 E 题)

## 课堂表现

今天进行了一些 01 背包题目对应的训练, 大部分同学课上掌握的越来越好。

## 课堂内容

### 最大约数和 (上周作业)

把问题转化为 01 背包: 第 i 个物品的体积为 i, 价值为 w[i] (w[i] 是 i 这个数的所有约数和)

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

using namespace std;

const int maxn = 1000 + 5;
int w[maxn], f[maxn];

void init() {
    for (int i = 2; i < maxn; ++i) {
        for (int j = 1; j < i; ++j) {
            if (i%j == 0) w[i] += j;
        }
    }
}
```

```

    }
}

int main()
{
    init();
    int n; cin >> n;
    for (int i = 2; i <= n; ++i) {
        for (int j = n; j >= i; --j) f[j] = max(f[j], f[j-i]+w[i]);
    }
    cout << f[n] << endl;
    return 0;
}

```

## 5 倍经验日

转化为 01 背包问题

第  $i$  个物品有 2 种情况:

1.  $c$  的体积  $wn$  的价值

2. 0 的体积  $ls$  的价值

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 1000 + 5;
LL f[maxn];

int main()
{
    int n, m; cin >> n >> m;
    while (n--) {
        int ls, wn, c; cin >> ls >> wn >> c;
        for (int i = m; i >= 0; --i) {
            if (i >= c) f[i] = max(f[i]+ls, f[i-c]+wn);
            else f[i] += ls;
        }
    }
    cout << 5*f[m] << endl;
    return 0;
}

```

## [USACO09OCT] Bessie's Weight Problem G

让重量和价值一比一, 想获得最大重量就是要获得最大价值

```
#include <bits/stdc++.h>
using namespace std;
const int maxn=50005;
int w[maxn],dp[maxn];
int main(){
    ios::sync_with_stdio(false);
    cin.tie(0);cout.tie(0);
    int n,m;cin>>m>>n;
    for(int i=1;i<=n;i++) cin>>w[i];
    for(int i=1;i<=n;i++)
        for(int j=m;j>=w[i];j--) dp[j]=max(dp[j],dp[j-w[i]]+w[i]);
    cout<<dp[m];
    return 0;
}
```

## [蓝桥杯 2021 省 AB] 磅码称重

$f[i][j]$ : 只用前  $i$  个砝码时, 能否称出来  $j$  这个重量

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

using namespace std;

const int maxn = 1e5 + 5;
bool f[105][maxn];

int main()
{
    f[0][0] = true;
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) {
        int x; cin >> x;
        for (int j = 0; j < maxn; ++j) {
            f[i][j] |= f[i-1][j];
            if (j+x < maxn) f[i][j+x] |= f[i-1][j];
            if (j-x >= 0) f[i][j-x] |= f[i-1][j];
            if (x-j >= 0) f[i][x-j] |= f[i-1][j];
        }
    }

    int res = 0;
    for (int i = 1; i < maxn; ++i) {
        if (f[n][i]) ++res;
    }
    cout << res << endl;
    return 0;
}
```

f[i]: 正好花完 j 元时有多少种方案

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

using namespace std;

const int maxn = 10000 + 5;
int f[maxn];

int main()
{
    f[0] = 1;

    int n, m; cin >> n >> m;
    while (n -- ) {
        int x; cin >> x;
        for (int j = m; j >= x; --j) f[j] += f[j-x];
    }

    cout << f[m] << endl;
    return 0;
}
```