

多重背包

人员

于家瑞、于霄龙、陶汇笙、郭栩睿 到课

上周作业检查

上周作业链接: <https://cppoj.kids123code.com/contest/2016>

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											
王向东老师周六十点半C++二维01背包+完全背包											
刷新											
#	用户名	姓名	编程分	时间	A	B	C	D	E	F	G
1	taohuisheng	陶汇笙	700	462	100	100	100	100	100	100	100
2	yujiarui	于家瑞	600	797	100	100	100	100	100		100
3	guoxurui	郭栩睿	600	4822	100	100	100	100	100	100	
4	yuxiaolong	于霄龙	555	366	100	100	100	100	100	55	
5	hongchenqi	洪晨祺	500	362	100	100	100	100	100	0	
6	hongchendong	洪晨栋	400	317	100		100	100	100		
7	wangenze	王思泽	90	0			90				

本周作业

<https://cppoj.kids123code.com/contest/2166> (课上讲了 A ~ D 这些题, 课后作业是 E F 题)

课堂表现

今天给同学们讲了 多重背包 这个知识点。

到目前为止, 一共讲了 01背包, 完全背包, 多重背包 这些内容, 这些内容的模板需要同学们熟练掌握, 哪个知识点不熟的, 同学们课下要多写几遍。

课堂内容

L 国的战斗之间谍 (上周作业)

f[i][j]: 当伪装能力为 i, 要工资为 j 时, 最多能得到多少资料

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

using namespace std;

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

int main()
{
```

```

int n, m, x; cin >> n >> m >> x;
while (n -- ) {
    int a, b, c; cin >> a >> b >> c;
    for (int i = m; i >= b; --i) {
        for (int j = x; j >= c; --j) f[i][j] = max(f[i][j], f[i-b][j-c]+a);
    }
}
cout << f[m][x] << endl;
return 0;
}

```

[NOIP 1996 提高组] 砝码称重

多重背包暴力版, 每个物品有 x 个, 可以跑 x 次 01 背包

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1000 + 5;
bool f[maxn];
int a[] = {0, 1, 2, 3, 5, 10, 20};

int main()
{
    f[0] = true;
    for (int i = 1; i <= 6; ++i) {
        int c = a[i], x; cin >> x;
        for (int j = 1; j <= x; ++j) {
            for (int k = 1000; k >= c; --k) f[k] |= f[k-c];
        }
    }

    int res = 0;
    for (int i = 1; i < maxn; ++i) res += f[i];
    cout << "Total=" << res << endl;
    return 0;
}

```

宝物筛选

多重背包二进制版, 用二进制进行优化

比如说有 100 个某物品, 可以认为有 1个、2个、4个、8个、16个、32个、37个 这些组

```

#include <bits/stdc++.h>

using namespace std;

```

```

const int maxn = 4e4 + 5;
int f[maxn];

int main()
{
    int n, W; cin >> n >> W;
    while (n -- ) {
        int v, w, m; cin >> v >> w >> m;
        int sum = 0;
        for (int i = 1; sum+i <= m; i *= 2) {
            for (int j = W; j >= i*w; --j) f[j] = max(f[j], f[j-i*w]+i*v);
            sum += i;
        }
        m -= sum;
        for (int j = W; j >= m*w; --j) f[j] = max(f[j], f[j-m*w]+m*v);
    }
    cout << f[W] << endl;
    return 0;
}

```

樱花

当 $p=0$ 时, 用完全背包进行更新

当 $p>0$ 时, 用多重背包进行更新

```

#include <bits/stdc++.h>

using namespace std;

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

int get_read() {
    int hh, mm; char c;
    cin >> hh >> c >> mm;
    return hh*60 + mm;
}

int main()
{
    int a = get_read(), b = get_read();
    int n; cin >> n;
    int m = b - a;
    while (n -- ) {
        int c, v, p; cin >> c >> v >> p;
        if (p == 0) {
            for (int i = c; i <= m; ++i) f[i] = max(f[i], f[i-c]+v);
        } else {
            for (int i = 1; i <= p; i *= 2) {
                for (int j = m; j >= i*c; --j) f[j] = max(f[j], f[j-i*c]+i*v);
            }
        }
    }
}

```

```

        p -= i;
    }
    for (int j = m; j >= p*c; --j) f[j] = max(f[j], f[j-p*c]+p*v);
}
}
cout << f[m] << endl;
return 0;
}

```

[USACO08MAR] River Crossing S

$f[i]$: 共运过去 i 只奶牛最小代价

$f[i] = \min(f[i-j]+p[j]+2*m) \ (1 \leq j \leq i)$

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 2500 + 5;
const LL inf = 0x3f3f3f3f3f3f3f3f;
LL w[maxn], p[maxn];
// f[i]: 共运过去 i 只奶牛最小代价

int main()
{
    int n, m; cin >> n >> m;
    for (int i = 1; i <= n; ++i) cin >> w[i], p[i] = p[i-1] + w[i];

    memset(f, 0x3f, sizeof(f));
    f[0] = 0;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= i; ++j) f[i] = min(f[i], f[i-j]+p[j]+2*m);
    }
    cout << f[n]-m << endl;
    return 0;
}

```