

# 贪心区间问题

---

## 人员

方俊喆、刘祺、夏硕承、牛同泽、柳力玮、高健桓、秦显森、赵熙羽、牟茗、徐浩然、齐振玮、夏志赫、武敬哲、谢亚锴 到课，辛帅辰 线上

## 作业检查

上周作业链接：<https://www.luogu.com.cn/contest/205677>

赵熙羽、夏志赫、秦显森、牟茗 做了 2 道题

辛帅辰、武敬哲、柳力玮、徐浩然、夏硕承、齐振玮 做了 1 道题

其他同学未做

## 作业

<https://www.luogu.com.cn/contest/208002> (D E 2 道题)

## 课堂表现

同学们听课都很认真，有些同学自己debug的能力弱一些，需要依赖老师来进行debug，这就是基础不太扎实的表现，需要课下进一步提高自己debug能力。

## 课堂内容

### P1803 凌乱的yyy / 线段覆盖

1. 按照右端点从小到大排序
2. 从左到右枚举一遍，如果与前面选的不冲突的话就选，与前面有冲突的话就不选

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

using namespace std;

const int maxn = 1e6 + 5;
struct node {
    int l, r;
} w[maxn];

bool cmp(node p, node q) {
    return p.r < q.r;
}

int main()
{
    int n; cin >> n;
```

```

for (int i = 1; i <= n; ++i) cin >> w[i].l >> w[i].r;
sort(w+1, w+n+1, cmp);

int last = -1, res = 0;
for (int i = 1; i <= n; ++i) {
    if (w[i].l >= last) {
        ++res;
        last = w[i].r;
    }
}

cout << res << endl;
return 0;
}

```

## U492914 区间覆盖

1. 按照左端点从小到大排序
2. 从左到右扫一遍，如果这个区间的左端点  $\leq$  限定值，就说明这个可选，对右端点进行取 max 操作 否则，说明不可直接选，先选前面的区间，再尝试选这个区间

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e5 + 5;
struct node {
    int l, r;
} w[maxn];

bool cmp(node p, node q) {
    return p.l < q.l;
}

int main()
{
    int s, t; cin >> s >> t;
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) cin >> w[i].l >> w[i].r;
    sort(w+1, w+n+1, cmp);

    int cnt = 0, last = s, r = -1e9;
    for (int i = 1; i <= n; ++i) {
        if (w[i].l <= last) {
            r = max(r, w[i].r);
            if (r >= t) {
                ++cnt; cout << cnt << endl; return 0;
            }
        } else {
            ++cnt; last = r; r = -1e9;
        }
    }
}

```

```

        if (w[i].l <= last) {
            r = max(r, w[i].r);
            if (r >= t) {
                ++cnt; cout << cnt << endl; return 0;
            }
        }
        else {
            cout << -1 << endl; return 0;
        }
    }
}

cout << -1 << endl;
return 0;
}

```

### U492913 区间分组

问题可以转化为：在每个时刻分别有多少个会议，最后看在哪个时刻有最多的会议数量即可

可以用 差分 的思想，这样每次对  $l \sim r$  区间  $+1$  的操作可以优化为  $O(1)$

```

#include <bits/stdc++.h>

using namespace std;

int main()
{
    map<int, int> mp;
    int n; cin >> n;
    while (n -- ) {
        int l, r; cin >> l >> r;
        mp[l]++, mp[r+1]--;
    }

    int sum = 0, maxx = 0;
    for (auto it : mp) {
        sum += it.second;
        maxx = max(maxx, sum);
    }
    cout << maxx << endl;
    return 0;
}

```

### P3817 小A的糖果

从左到右扫一遍，每次让右边的变即可

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

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int w[maxn];

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

    LL res = 0;
    for (int i = 2; i <= n; ++i) {
        int t = w[i - 1] + w[i] - x;
        if (t > 0) {
            res += t;
            w[i] -= t;
            if (w[i] < 0) w[i] = 0;
        }
    }
    cout << res << endl;
    return 0;
}
```