

1주차 연습문제 풀이

STOP USING MONEY BOJ 14674

- 가성비 내림차순, 가격 오름차순, 번호 오름차순 정렬

```
struct game {  
    int i, c, h;  
};
```

```
int n, k; cin >> n >> k; vector<game> g(n);  
for (auto &x: g) cin >> x.i >> x.c >> x.h;  
sort(g.begin(), g.end(), comp);
```

```
bool comp(const game &a, const game &b) {  
    if ((double)a.h / a.c != (double)b.h / b.c) {  
        return (double)a.h / a.c > (double)b.h / b.c;  
    }  
    if (a.c != b.c) return a.c < b.c;  
    return a.i < b.i;  
}
```

STOP USING MONEY BOJ 14674

- 실수를 다루는 것은 매우 위험함

```
struct game {  
    long long i, c, h;  
};
```

```
int n, k; cin >> n >> k; vector<game> g(n);  
for (auto &x: g) cin >> x.i >> x.c >> x.h;  
sort(g.begin(), g.end(), comp);
```

```
bool comp(const game &a, const game &b) {  
    if (a.h * b.c != b.h * a.c) {  
        return a.h * b.c > b.h * a.c;  
    }  
    if (a.c != b.c) return a.c < b.c;  
    return a.i < b.i;  
}
```

회문수 BOJ 30446

- $1 \leq n \leq 10^{10}$

```
typedef long long ll;
```

```
ll n; cin >> n;
```

```
bool palindrome(ll x) {  
    string n = to_string(x);  
    int l = 0, r = n.size() - 1;  
    while (l < r) {  
        if (n[l++] != n[r--]) return false;  
    }  
    return true;  
}
```

```
ll ans = 0;  
for (ll i = 1; i <= n; i++) {  
    if (palindrome(i)) ans++;  
}  
cout << ans << '\n';
```

$O(n \log_{10} n)$

시간 초과

회문수 BOJ 30446

- 회문수의 특징을 이용
- $O(\sqrt{n} \log_{10} n)$
- `std::to_string`
- `std::stoll`

그리디 알고리즘

8만원을 지폐를 최소로 사용해서 나타내기



최적해 - 4장 사용



다른 예시 - 8장 사용

그리디 알고리즘

- 방금 무의식적으로 사용한 방법이 바로 그리디 알고리즘



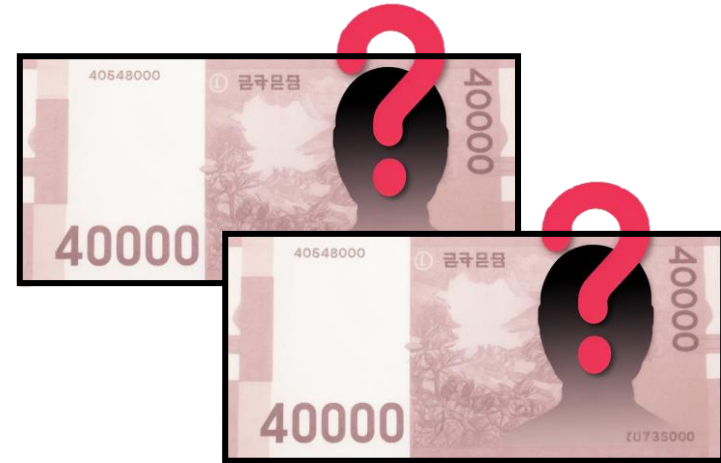
만약 4만원권 지폐가 생긴다면



그리디가 항상 최적해를 보장하지는 않음



그리디 알고리즘 - 4장 사용



최적해 - 2장 사용

그리디는 문제를 푸는 방법 중 하나

- 생각한 방법이 항상 올바른지 증명하는 것이 필요
- 만약 이미 푼 다른 문제로 환원할 수 있다면 시간을 절약할 수 있음

풀이에 확신이 들지 않는다면?

- 다른 방법으로 접근을 시도
- 틀릴 각오와 함께 제출 – 대회에서는 **비추천**

우유 축제 BOJ 14720

우유 축제 BOJ 14720

“지금 마셔야 할 종류를 파는 가게면 사마시고, 그렇지 않으면 지나친다”

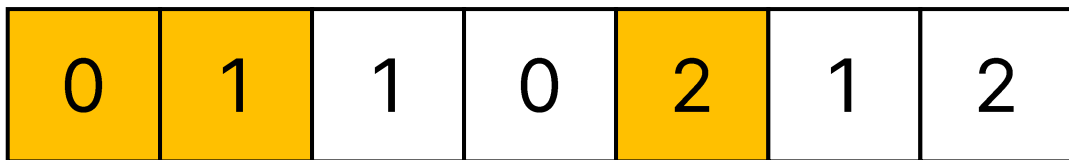
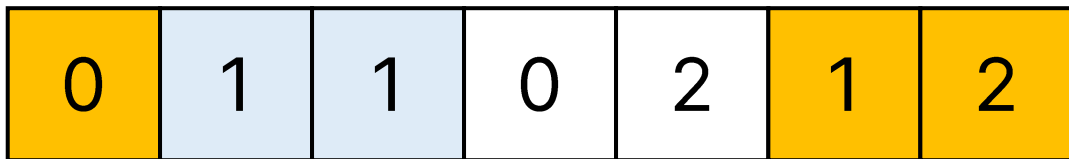
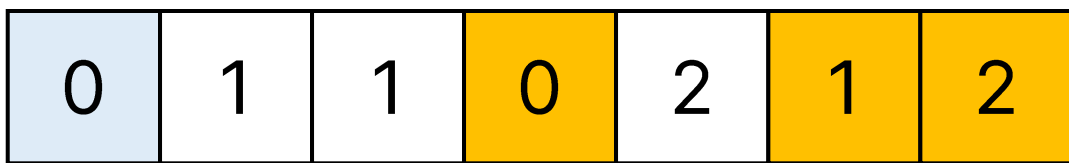
어떻게 증명할까?

우유 축제 BOJ 14720

- 지금 마셔야 할 종류를 파는 가게일 경우 – 사마시거나 지나칠 수 있음
- 지금 마셔야 할 종류를 팔지 않는 가게일 경우 – 지나칠 수밖에 없음
- 만약 지금 마셔야 할 종류를 파는 가게를 지나쳤을 경우, 사마셨을 때보다 더 많이 마실 수 없음을 보이면 됨

우유 축제 BOJ 14720

- 만약 지금 마셔야 할 종류를 파는 가게를 지나친 경우, 지나치지 않고도 최적해임



□ 사마실 수 있지만 지나친 가게

■ 사마신 가게

□ 사마실 수 없는 가게

어디까지 증명해야 하나요?

- 명확한 기준은 없지만 본인이 확신이 들 정도로 증명하는 것이 최선
- 사람에 따라서 확신이 들 정도로 증명해도 틀릴 수 있음
- 확신이 들 정도로 증명했지만 자주 틀릴 경우, 기준을 높일 필요가 있음

회의실 배정 BOJ 1931

회의실 배정 BOJ 1931

- $1 \leq N \leq 10^5$
- 제한을 보고 풀이의 시간복잡도를 유추하는 것도 방법
- $O(N \log N)$ 풀이를 작성하면 통과할 수 있다는 사실을 알 수 있음

특정 기준으로 정렬한 후 순차적으로 회의실을 배정하면 되지 않을까?

회의실 배정 BOJ 1931

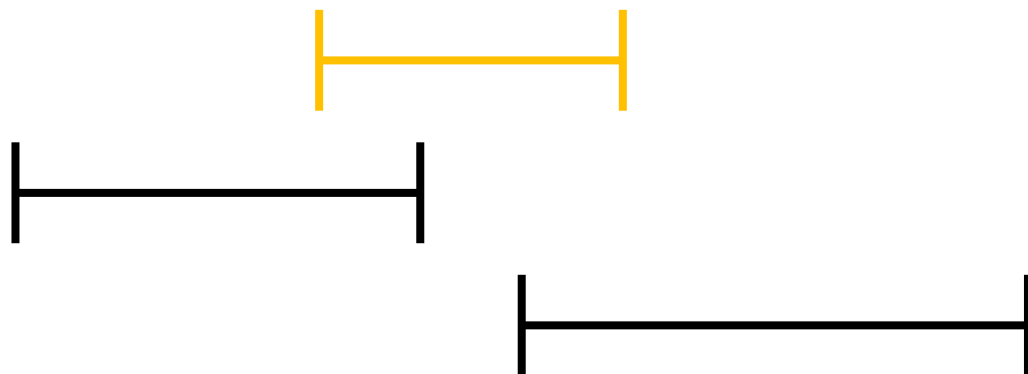
- 정렬 기준을 몇가지 생각해볼 수 있음
- 가장 짧게 진행되는 회의부터 배정?
- 가장 빨리 시작하는 회의부터 배정?
- 가장 빨리 종료되는 회의부터 배정?
- 올바르지 않은 풀이는 빠르게 제외하는 것이 유리

회의실 배정 BOJ 1931

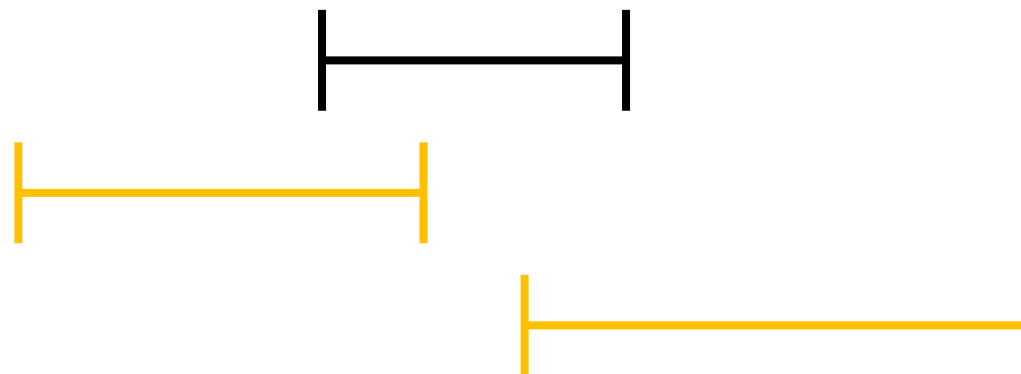
입력

| |
|------|
| 3 |
| 1 5 |
| 4 7 |
| 6 11 |

- 가장 짧게 진행되는 회의부터 배정?



가장 짧게 진행되는 회의부터 배정 - 1개



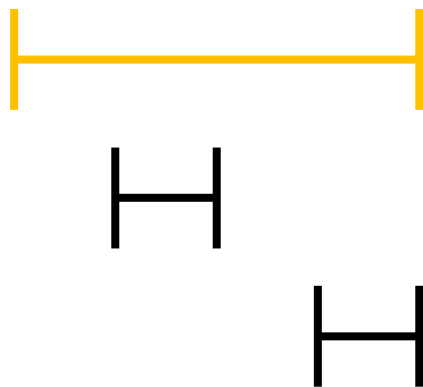
최적해 - 2개

회의실 배정 BOJ 1931

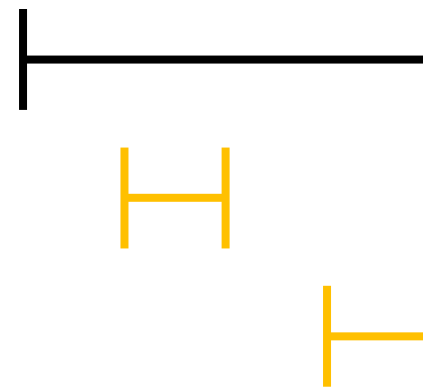
입력

| |
|-----|
| 3 |
| 1 5 |
| 2 3 |
| 4 5 |

- 가장 빨리 시작하는 회의부터 배정?



가장 빨리 시작하는 회의부터 배정 – 1개



최적해 – 2개

회의실 배정 BOJ 1931

- 가장 빨리 종료되는 회의부터 배정?

뭔가 맞는 것 같음 → 증명해보자!

회의실 배정 BOJ 1931

- 가장 빨리 종료되는 회의를 배정하지 않았을 때 최적해가 존재한다고 가정
- 배정한 회의 중에서 가장 빨리 종료되는 회의 대신 전체 회의 중에서 가장 빨리 종료되는 회의로 교체 가능
- 즉, 가장 빨리 종료되는 회의를 배정해도 최적해
- 가장 빨리 종료되는 회의를 배정하면 남은 회의 중에서 기존 회의와 겹치는 회의를 제외하고 같은 작업 반복

팁

- 문제를 풀 때 생각한 풀이가 올바른지는 증명하기 전까지 알 수 없음
- 생각한 풀이가 틀린지 확인하기 위해서는 반례 하나면 충분함
- 풀이가 떠올랐다면 먼저 반례를 찾으려고 노력하기
- 반례를 못 찾았다면 증명을 통해 풀이가 올바름을 보이기

연습문제

| | |
|--------------|------------------|
| <u>11000</u> | : 강의실 배정 |
| <u>11047</u> | : 동전 0 |
| <u>11399</u> | : ATM |
| <u>14464</u> | : 소가 길을 건너간 이유 4 |
| <u>25379</u> | : 피하자 |
| <u>25943</u> | : 양팔저울 |
| <u>27919</u> | : UDPC 파티 |
| <u>28305</u> | : 세미나 배정 |
| <u>31231</u> | : 마카롱카마 |
| <u>31410</u> | : 제독 작전 |