

CSES PLAN I

CSES PLAN I

因為我跟 peienwu 被揍爛了，於是我們決定寫CSES來增進自己的實力
[peienwu CSES補完計畫](https://hackmd.io/@peienwu/cses#CSES-%E8%A3%9C%E5%AE%8C%E8%A8%88%E7%95%AB)

目錄

[CSES PLAN I](https://hackmd.io/@thanksone/CSESPLANI)

- Introductory Problems
- Sorting and Searching
- Dynamic Programming
- Graph Algorithms

[CSES PLAN II](https://hackmd.io/@thanksone/CSESPLANII)

- Range Queries
- Tree Algorithms
- Mathematics
- String Algorithms
- Geometry

[CSES PLAN III](https://hackmd.io/@thanksone/CSESPLANIII)

- Advanced Techniques

[CSES PLAN IV](https://hackmd.io/@thanksone/CSESPLANIV)

- Additional Problems

Introductory Problems

[Weird Algorithm](https://cses.fi/problemset/task/1068)

暴力

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int n;
7     cin >> n;
8     cout << n << " ";
9     while(n != 1){
10         if(n & 1) n = 3 * n + 1;
11         else n /= 2;
12         cout << n << " ";
13     }
14     return 0;
15 }
```

Missing Number (<https://cses.fi/problemset/task/1083>)

暴力

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int n, x, sum = 0, tot;
7     cin >> n;
8     tot = n * (n + 1) / 2;
9     while(--n){
10         cin >> x;
11         sum += x;
12     }
13     cout << tot - sum;
14     return 0;
15 }
```

Repetitions (<https://cses.fi/problemset/task/1069>)

暴力

```

1 #include <bits/stdc++.h>
2 using namespace std;
3
4 signed main(){
5     string S;
6     int sum = 0, ans = 0;
7     char lst = '0';
8     cin >> S;
9     for(char s : S){
10         if(s == lst) sum++;
11         else{
12             lst = s;
13             sum = 1;
14         }
15         ans = max(ans, sum);
16     }
17     cout << ans;
18     return 0;
19 }
```

Increasing Array (<https://cses.fi/problemset/task/1094>)

暴力

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int n, x, lst = 0, ans = 0;
7     cin >> n;
8     for(int i = 0; i < n; i++){
9         cin >> x;
10        if(x < lst) ans += lst - x;
11        else lst = x;
12    }
13    cout << ans;
14    return 0;
15 }
```

Permutations (<https://cses.fi/problemset/task/1070>)

暴力

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<bool, 1000004> vis;
4 signed main(){
5     int n;
6     cin >> n;
7     if(n < 4 && n > 1){
8         cout << "NO SOLUTION";
9         return 0;
10    }
11    for(int i = n - !(n & 1); i >= 1; i -= 2){
12        cout << i << " ";
13    }
14    for(int i = n - (n & 1); i >= 2; i -= 2){
15        cout << i << " ";
16    }
17    return 0;
18 }
```

Number Spiral (<https://cses.fi/problemset/task/1071>)

數學

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int t, x, y;
7     cin >> t;
8     while(t--){
9         cin >> x >> y;
10        if(x > y){
11            if(x & 1){
12                cout << (x - 1) * (x - 1) + y;
13            }else{
14                cout << (x - 1) * (x - 1) + (2 * x - y);
15            }
16        }else{
17            if(y & 1){
18                cout << (y - 1) * (y - 1) + (2 * y - x);
19            }else{
20                cout << (y - 1) * (y - 1) + x;
21            }
22        }
23        cout << "\n";
24    }
25    return 0;
26 }
```

Two Knights (<https://cses.fi/problemset/task/1072>)

數學

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int n;
7     cin >> n;
8     for(int k = 1; k <= n; k++){
9         /*
10            all : k^2(k^2 - 1) / 2
11            no : 8(k - 2)(k - 1) / 2
12            = 4(k - 2)(k - 1)
13        */
14        cout << k * k * (k * k - 1) / 2 - 4 * (k - 2) * (k - 1) << "\n";
15    }
16    return 0;
17 }
```

Two Sets (<https://cses.fi/problemset/task/1092>)

暴力

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<bool, 1000004> vis;
5 void bag(int n){
6     int tot = n * (n + 1) / 2, sum = 0, cnt = 0;
7     if(tot & 1){
8         cout << "NO";
9         return;
10    }
11    cout << "YES\n";
12    for(int i = n; i > 0; i--){
13        if(sum < tot / 2){
14            sum += i;
15            vis[i] = 1;
16            cnt++;
17        }
18        if(sum > tot / 2){
19            sum -= i;
20            vis[i] = 0;
21            cnt--;
22        }
23    }
24    cout << cnt << "\n";
25    for(int i = 1; i <= n; i++){
26        if(vis[i]) cout << i << " ";
27    }
28    cout << n - cnt << "\n";
29    cout << "\n";
30    for(int i = 1; i <= n; i++){
31        if(!vis[i]) cout << i << " ";
32    }
33 }
34 signed main(){
35     int n;
36     cin >> n;
37     bag(n);
38     return 0;
39 }
```

Bit Strings (<https://cses.fi/problemset/task/1617>)

暴力

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 int go(int n){
5     int k = 1;
6     while(n--){
7         k <= 1;
8         k %= mod;
9     }
10    return k;
11 }
12 signed main(){
13     int n;
14     cin >> n;
15     cout << go(n);
16     return 0;
17 }
```

Trailing Zeros (<https://cses.fi/problemset/task/1618>).

暴力

```

1 #include <bits/stdc++.h>
2 using namespace std;
3
4 signed main(){
5     int n, f = 5, ans = 0;
6     cin >> n;
7     for(int i = 1; i < 13; i++){
8         ans += n / f;
9         f *= 5;
10    }
11    cout << ans;
12    return 0;
13 }
```

Coin Piles (<https://cses.fi/problemset/task/1754>)

數學

```

1 #include <bits/stdc++.h>
2 using namespace std;
3
4 signed main(){
5     int t, a, b;
6     cin >> t;
7     while(t--){
8         cin >> a >> b;
9         if((a + b) % 3 == 0 && a >= b / 2 + b % 2 && b >= a / 2 + a % 2) cout
10            else cout << "NO\n";
11    }
12    return 0;
13 }
```

Palindrome Reorder (<https://cses.fi/problemset/task/1755>)

暴力

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 string S;
4 array<int, 26> C;
5 array<char, 1000004> ans;
6 void go(int n){
7     int p = 0;
8     for(int i = 0; i < n / 2; i++){
9         while(C[p] < 2) p++;
10        ans[i] = ans[n - 1 - i] = p + 'A';
11        C[p] -= 2;
12    }
13    if(n & 1){
14        for(int i = 0; i < 26; i++){
15            if(C[i]) ans[n / 2] = i + 'A';
16        }
17    }
18    for(int i = 0; i < n; i++){
19        cout << ans[i];
20    }
21 }
22 signed main(){
23     cin >> S;
24     int cnt = 0, n;
25     n = S.size();
26     for(char s : S){
27         C[s - 'A']++;
28     }
29     for(int c : C){
30         cnt += c & 1;
31     }
32     if(cnt - (n & 1)) cout << "NO SOLUTION";
33     else go(n);
34     return 0;
35 }
```

Gray Code (<https://cses.fi/problemset/task/2205>)

暴力

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 vector<string> S;
5 signed main(){
6     int n;
7     cin >> n;
8     S.pb("0");
9     S.pb("1");
10    for(int i = 2; i < (1 << n); i <= 1){
11        for(int j = i - 1; j >= 0; j--){
12            S.pb(S[j]);
13        }
14        for(int j = 0; j < i; j++){
15            S[j] = "0" + S[j];
16        }
17        for(int j = i; j < 2 * i; j++){
18            S[j] = "1" + S[j];
19        }
20    }
21    for(string s : S){
22        cout << s << "\n";
23    }
24    return 0;
25 }
```

Tower of Hanoi (<https://cses.fi/problemset/task/2165>)

遞迴

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 void go(int n, int s, int t){
4     int mid = 6 - s - t;
5     if(!n) return;
6     go(n - 1, s, mid);
7     cout << s << " " << t << "\n";
8     go(n - 1, mid, t);
9 }
10 signed main(){
11     int n;
12     cin >> n;
13     cout << (1 << n) - 1 << "\n";
14     go(n, 1, 3);
15     return 0;
16 }
```

Creating Strings (<https://cses.fi/problemset/task/1622>)

暴力

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 26> C;
5 vector<string> SS;
6 signed main(){
7     string S;
8     cin >> S;
9     for(char s : S){
10         C[s - 'a']++;
11     }
12     S.clear();
13     for(int i = 0; i < 26; i++){
14         while(C[i]){
15             S.pb(i + 'a');
16             C[i]--;
17         }
18     }
19     SS.pb(S);
20     while(next_permutation(S.begin(), S.end())){
21         SS.pb(S);
22     }
23     cout << SS.size() << "\n";
24     for(string s : SS){
25         cout << s << "\n";
26     }
27     return 0;
28 }
```

Apple Division (<https://cses.fi/problemset/task/1623>)

暴力

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 20> P;
5 int go(int n){
6     int a, b, ans = 2e10;
7     for(int i = 0; i < 1 << n; i++){
8         a = b = 0;
9         for(int j = 0; j < n; j++){
10             if(i & (1 << j)) a += P[j];
11             else b += P[j];
12         }
13         ans = min(ans, abs(a - b));
14     }
15     return ans;
16 }
17 signed main(){
18     int n;
19     cin >> n;
20     for(int i = 0; i < n; i++){
21         cin >> P[i];
22     }
23     cout << go(n);
24     return 0;
25 }
```

Chessboard and Queens (<https://cses.fi/problemset/task/1624>)

暴力

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<array<int, 12>, 12> C;
4 void update(int x, int y, int v){
5     for(int i = 1; i <= 8; i++){
6         C[x][i] += v;
7         C[i][y] += v;
8     }
9     for(int i = 1; i < 8; i++){
10        if(x + i > 8 || y + i > 8) break;
11        C[x + i][y + i] += v;
12    }
13    for(int i = 1; i < 8; i++){
14        if(x + i > 8 || y - i < 1) break;
15        C[x + i][y - i] += v;
16    }
17    for(int i = 1; i < 8; i++){
18        if(x - i < 1 || y + i > 8) break;
19        C[x - i][y + i] += v;
20    }
21    for(int i = 1; i < 8; i++){
22        if(x - i < 1 || y - i < 1) break;
23        C[x - i][y - i] += v;
24    }
25 }
26 int dfs(int n){
27     int ans = 0;
28     if(!n) return 1;
29     for(int i = 1; i <= 8; i++){
30         if(C[n][i]) continue;
31         update(n, i, 1);
32         ans += dfs(n - 1);
33         update(n, i, -1);
34     }
35     return ans;
36 }
37 signed main(){
38     char c;
39     for(int i = 1; i <= 8; i++){
40         for(int j = 1; j <= 8; j++){
41             cin >> c;
42             if(c == '*') C[i][j]++;
43         }
44     }
45     cout << dfs(8);
46     return 0;
47 }
48 }
```

Digit Queries (<https://cses.fi/problemset/task/2431>)

數學

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 int pwr(int x, int k){
5     int v = 1;
6     for(int i = 1; i <= 18; i <= 1){
7         if(k & i) v *= x;
8         x *= x;
9     }
10    return v;
11 }
12 int dig(int k){
13     int n = 1, ans;
14     while(k > 9 * n * pwr(10, n - 1)){
15         k -= 9 * n * pwr(10, n - 1);
16         n++;
17     }
18     ans = pwr(10, n - 1) + (k - 1) / n;
19     k = (k - 1) % n;
20     ans = (ans / pwr(10, n - k - 1)) % 10;
21     return ans;
22 }
23 signed main(){
24     int q, k;
25     cin >> q;
26     while(q--){
27         cin >> k;
28         cout << dig(k) << "\n";
29     }
30     return 0;
31 }
```

Grid Paths (<https://cses.fi/problemset/task/1625>)

暴力 DFS

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<char, 48> S;
4 array<array<bool, 9>, 9> vis;
5 int ans = 0;
6 void dfs(int x, int y, int s){
7     if(x < 1 || y < 1 || x > 7 || y > 7 || vis[x][y]) return;
8     if(x == 1 && y == 7){
9         if(s == 48) ans++;
10        return;
11    }
12    if(!vis[x - 1][y] && !vis[x + 1][y] && vis[x][y - 1] && vis[x][y + 1]) ret
13    if(vis[x - 1][y] && vis[x + 1][y] && !vis[x][y - 1] && !vis[x][y + 1]) ret
14    vis[x][y] = 1;
15    if(S[s] == '?' || S[s] == 'L') dfs(x - 1, y, s + 1);
16    if(S[s] == '?' || S[s] == 'R') dfs(x + 1, y, s + 1);
17    if(S[s] == '?' || S[s] == 'U') dfs(x, y - 1, s + 1);
18    if(S[s] == '?' || S[s] == 'D') dfs(x, y + 1, s + 1);
19    vis[x][y] = 0;
20}
21 signed main(){
22     for(int i = 0; i < 48; i++){
23         cin >> S[i];
24     }
25     for(int i = 1; i <= 7; i++){
26         vis[0][i] = 1;
27         vis[i][0] = 1;
28         vis[8][i] = 1;
29         vis[i][8] = 1;
30     }
31     dfs(1, 1, 0);
32     cout << ans;
33     return 0;
34 }
```

Sorting and Searching

Distinct Numbers (<https://cses.fi/problemset/task/1621>)

排序

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> X;
4 signed main(){
5     int n, lst = 0, ans = 0;
6     cin >> n;
7     for(int i = 0; i < n; i++){
8         cin >> X[i];
9     }
10    sort(X.begin(), X.begin() + n);
11    for(int i = 0; i < n; i++){
12        if(X[i] != lst) ans++;
13        lst = X[i];
14    }
15    cout << ans;
16    return 0;
17 }
```

Apartments (<https://cses.fi/problemset/task/1084>)

排序 雙指針

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> A, B;
4 signed main(){
5     int n, m, k, p = 0, ans = 0;
6     cin >> n >> m >> k;
7     for(int i = 0; i < n; i++){
8         cin >> A[i];
9     }
10    for(int i = 0; i < m; i++){
11        cin >> B[i];
12    }
13    sort(A.begin(), A.begin() + n);
14    sort(B.begin(), B.begin() + m);
15    for(int i = 0; i < n; i++){
16        while(p < m && abs(B[p] - A[i]) > k && B[p] < A[i] + k) p++;
17        if(p < m && abs(B[p] - A[i]) <= k){
18            ans++;
19            p++;
20        }
21    }
22    cout << ans;
23    return 0;
24 }
```

Ferris Wheel (<https://cses.fi/problemset/task/1090>)

排序 雙指針

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> P;
4 signed main(){
5     int n, x, p, ans = 0;
6     cin >> n >> x;
7     for(int i = 0; i < n; i++){
8         cin >> P[i];
9     }
10    sort(P.begin(), P.begin() + n);
11    p = n - 1;
12    for(int i = 0; i <= p; i++){
13        while(i < p && P[p] + P[i] > x){
14            p--;
15            ans++;
16        }
17        if(i >= p || (i < p && P[p] + P[i] <= x)){
18            p--;
19            ans++;
20        }
21    }
22    cout << ans;
23    return 0;
24 }
```

Concert Tickets (<https://cses.fi/problemset/task/1091>)

Set

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 multiset<int> H;
4 signed main(){
5     int n, m, h, t;
6     cin >> n >> m;
7     for(int i = 0; i < n; i++){
8         cin >> h;
9         H.insert(h);
10    }
11    for(int i = 0; i < m; i++){
12        cin >> t;
13        if(H.upper_bound(t) == H.begin()) cout << "-1\n";
14        else{
15            cout << *--H.upper_bound(t) << "\n";
16            H.erase(--H.upper_bound(t));
17        }
18    }
19    return 0;
20 }
```

Restaurant Customers (<https://cses.fi/problemset/task/1619>)

排序

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 vector<pii> P;
8 signed main(){
9     int n, l, r, now = 0, ans = 0;
10    cin >> n;
11    for(int i = 0; i < n; i++){
12        cin >> l >> r;
13        P.pb({l, 1});
14        P.pb({r, -1});
15    }
16    sort(P.begin(), P.end());
17    for(pii p : P){
18        now += p.ss;
19        ans = max(ans, now);
20    }
21    cout << ans;
22    return 0;
23 }
```

Movie Festival (<https://cses.fi/problemset/task/1629>)

排序

```

1 #include <bits/stdc++.h>
2 #define pii pair<int, int>
3 #define ff first
4 #define ss second
5 using namespace std;
6 bool cmp(pii a, pii b){
7     if(a.ss != b.ss) return a.ss < b.ss;
8     return a.ff < b.ff;
9 }
10 array<pii, 200004> P;
11 signed main(){
12     int n, l, r, ans = 0;
13     cin >> n;
14     for(int i = 0; i < n; i++){
15         cin >> l >> r;
16         P[i] = {l, r};
17     }
18     sort(P.begin(), P.begin() + n, cmp);
19     l = 0;
20     for(pii p : P){
21         if(p.ff >= l){
22             ans++;
23             l = p.ss;
24         }
25     }
26     cout << ans;
27     return 0;
28 }
```

Sum of Two Values (<https://cses.fi/problemset/task/1640>)

Set Map

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 set<int> A;
4 map<int, int> M;
5 signed main(){
6     int n, x, a;
7     bool ans = 0;
8     cin >> n >> x;
9     for(int i = 1; i <= n; i++){
10         cin >> a;
11         if(A.lower_bound(x - a) != A.end()){
12             if(*A.lower_bound(x - a) == x - a){
13                 ans = 1;
14                 cout << i << " " << M[x - a];
15                 break;
16             }
17         }
18         A.insert(a);
19         M[a] = i;
20     }
21     if(!ans) cout << "IMPOSSIBLE";
22     return 0;
23 }
```

Maximum Subarray Sum (<https://cses.fi/problemset/task/1643>)

Greedy

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4
5 signed main(){
6     int n, ans = -1e18, sum = 0, x;
7     cin >> n;
8     for(int i = 0; i < n; i++){
9         cin >> x;
10        sum += x;
11        ans = max(ans, sum);
12        if(sum < 0) sum = 0;
13    }
14    cout << ans;
15    return 0;
16 }
```

Stick Lengths (<https://cses.fi/problemset/task/1074>)

二分搜

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> P;
5 int cost(int n, int k){
6     int sum = 0;
7     for(int i = 0; i < n; i++){
8         sum += labs(k - P[i]);
9     }
10    return sum;
11 }
12 int BS(int n){
13     int l = 1, r = 1e9, mid;
14     while(l != r){
15         mid = (l + r) / 2;
16         if(cost(n, mid) < cost(n, mid + 1)) r = mid;
17         else l = mid + 1;
18     }
19     return cost(n, l);
20 }
21 signed main(){
22     int n;
23     cin >> n;
24     for(int i = 0; i < n; i++){
25         cin >> P[i];
26     }
27     cout << BS(n);
28     return 0;
29 }
```

Missing Coin Sum (<https://cses.fi/problemset/task/2183>)

排序

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> X;
5 signed main(){
6     int n, sum = 0;
7     cin >> n;
8     for(int i = 0; i < n; i++){
9         cin >> X[i];
10    }
11    sort(X.begin(), X.begin() + n);
12    for(int i = 0; i < n; i++){
13        if(X[i] > sum + 1){
14            cout << sum + 1;
15            return 0;
16        }
17        sum += X[i];
18    }
19    cout << sum + 1;
20    return 0;
21 }
```

Collecting Numbers (<https://cses.fi/problemset/task/2216>)

暴力

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> X;
4 signed main(){
5     int n, x, ans = 1;
6     cin >> n;
7     for(int i = 0; i < n; i++){
8         cin >> x;
9         X[x] = i;
10    }
11    for(int i = 1; i <= n; i++){
12        ans += X[i] < X[i - 1];
13    }
14    cout << ans;
15    return 0;
16 }
```

Collecting Numbers II (<https://cses.fi/problemset/task/2217>)

暴力

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> X, P;
4 int check(int a, int b){
5     int sum = 0;
6     if(X[a - 1] > X[a]) sum++;
7     if(X[a + 1] < X[a]) sum++;
8     if(X[b - 1] > X[b]) sum++;
9     if(X[b + 1] < X[b]) sum++;
10    if(a - b == 1 && X[a] < X[b]) sum--;
11    if(b - a == 1 && X[b] < X[a]) sum--;
12    return sum;
13 }
14 signed main(){
15     int n, m, a, b, x, ans = 1;
16     cin >> n >> m;
17     X[n + 1] = n + 1;
18     for(int i = 1; i <= n; i++){
19         cin >> x;
20         P[i] = x;
21         X[x] = i;
22     }
23     for(int i = 1; i <= n; i++){
24         ans += X[i] < X[i - 1];
25     }
26     while(m--){
27         cin >> a >> b;
28         swap(P[a], P[b]);
29         a = P[a], b = P[b];
30         ans -= check(a, b);
31         swap(X[a], X[b]);
32         ans += check(a, b);
33         cout << ans << "\n";
34     }
35     return 0;
36 }
```

Playlist (<https://cses.fi/problemset/task/1141>)

Set

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 set<int> S;
4 array<int, 200004> K;
5 signed main(){
6     int n, ans = 0, p = 0;
7     cin >> n;
8     for(int i = 0; i < n; i++){
9         cin >> K[i];
10        while(S.find(K[i]) != S.end()){
11            S.erase(K[p++]);
12        }
13        S.insert(K[i]);
14        ans = max(ans, i - p + 1);
15    }
16    cout << ans;
17    return 0;
18 }
```

Towers (<https://cses.fi/problemset/task/1073>).

Set

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 multiset<int> T;
4 signed main(){
5     int n, k;
6     cin >> n;
7     for(int i = 0; i < n; i++){
8         cin >> k;
9         if(T.upper_bound(k) != T.end()){
10            T.erase(T.upper_bound(k));
11        }
12        T.insert(k);
13    }
14    cout << T.size();
15    return 0;
16 }
```

Traffic Lights (<https://cses.fi/problemset/task/1163>)

Set

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 set<int> T;
4 multiset<int> dis;
5 signed main(){
6     int x, n, p, l, r;
7     cin >> x >> n;
8     dis.insert(x);
9     T.insert(0);
10    T.insert(x);
11    for(int i = 0; i < n; i++){
12        cin >> p;
13        l = 0, r = x;
14        auto it = T.upper_bound(p);
15        r = *it;
16        l = *--it;
17        dis.erase(dis.find(r - l));
18        dis.insert(r - p);
19        dis.insert(p - l);
20        T.insert(p);
21        cout << *--dis.end() << " ";
22    }
23    return 0;
24 }
```

Josephus Problem I (<https://cses.fi/problemset/task/2162>)

BIT

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> BIT;
4 void update(int p, int x){
5     for(; p < 200004; p += p & -p){
6         BIT[p] += x;
7     }
8 }
9 int query(int p){
10    int sum = 0;
11    for(; p > 0; p -= p & -p){
12        sum += BIT[p];
13    }
14    return sum;
15 }
16 int find(int x){
17    int p = 0, sum = 0;
18    for(int i = (1 << 17); i > 0; i >= 1){
19        if(p + i < 200004 && sum + BIT[p + i] < x){
20            p += i;
21            sum += BIT[p];
22        }
23    }
24    return p + 1;
25 }
26 signed main(){
27     int n, now, p;
28     cin >> n;
29     p = n;
30     now = 0;
31     for(int i = 1; i <= n; i++){
32         update(i, 1);
33     }
34     while(p > 1){
35         now = find((query(now) + 1) % query(n) + 1);
36         cout << now << " ";
37         update(now, -1);
38         p--;
39     }
40     cout << find(1);
41     return 0;
42 }
```

Josephus Problem II (<https://cses.fi/problemset/task/2163>)

BIT

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> BIT;
4 void update(int p, int x){
5     for(; p < 200004; p += p & -p){
6         BIT[p] += x;
7     }
8 }
9 int query(int p){
10    int sum = 0;
11    for(; p > 0; p -= p & -p){
12        sum += BIT[p];
13    }
14    return sum;
15 }
16 int find(int x){
17    int p = 0, sum = 0;
18    for(int i = (1 << 17); i > 0; i >>= 1){
19        if(p + i < 200004 && sum + BIT[p + i] < x){
20            p += i;
21            sum += BIT[p];
22        }
23    }
24    return p + 1;
25 }
26 signed main(){
27     int n, now, p, k;
28     cin >> n >> k;
29     p = n;
30     now = 0;
31     for(int i = 1; i <= n; i++){
32         update(i, 1);
33     }
34     while(p > 1){
35         now = find((query(now) + k) % query(n) + 1);
36         cout << now << " ";
37         update(now, -1);
38         p--;
39     }
40     cout << find(1);
41     return 0;
42 }
```

Nested Ranges Check (<https://cses.fi/problemset/task/2168>)

排序 BIT


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 struct rng{
8     int l, r, t;
9 };
10 array<int, 200004> BIT, ans;
11 array<rng, 200004> R;
12 vector<pii> X, Y;
13 bool cmp(rng a, rng b){
14     if(a.l != b.l) return a.l < b.l;
15     return a.r > b.r;
16 }
17 void reset(){
18     for(int &b : BIT) b = 0;
19 }
20 void update(int p){
21     for(; p < 200004; p += p & -p){
22         BIT[p]++;
23     }
24 }
25 int query(int p){
26     int sum = 0;
27     for(; p > 0; p -= p & -p){
28         sum += BIT[p];
29     }
30     return sum;
31 }
32 signed main(){
33     int n, x, y, lst = 0, cnt = 0;
34     cin >> n;
35     for(int i = 0; i < n; i++){
36         cin >> x >> y;
37         X.pb({x, i});
38         Y.pb({y, i});
39         R[i].t = i;
40     }
41     sort(X.begin(), X.end());
42     sort(Y.begin(), Y.end());
43     for(pii p : X){
44         if(p.ff != lst) cnt++;
45         R[p.ss].l = cnt;
46         lst = p.ff;
47     }
48     lst = cnt = 0;
49     for(pii p : Y){
50         if(p.ff != lst) cnt++;
51         R[p.ss].r = cnt;
52         lst = p.ff;
53     }
54     sort(R.begin(), R.begin() + n, cmp);
55     for(int i = n - 1; i >= 0; i--){
56         ans[R[i].t] = query(R[i].r);
57         update(R[i].r);
58     }
59     for(int i = 0; i < n; i++){
60         cout << !ans[i] << " ";
61         ans[i] = 0;
62     }
63     cout << "\n";
64     reset();
65     for(int i = 0; i < n; i++){
```

```
66         ans[R[i].t] = query(200001) - query(R[i].r - 1);
67         update(R[i].r);
68     }
69     for(int i = 0; i < n; i++){
70         cout << !ans[i] << " ";
71     }
72     return 0;
73 }
```

Nested Ranges Count (<https://cses.fi/problemset/task/2169>)

排序 BIT


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 struct rng{
8     int l, r, t;
9 };
10 array<int, 200004> BIT, ans;
11 array<rng, 200004> R;
12 vector<pii> X, Y;
13 bool cmp(rng a, rng b){
14     if(a.l != b.l) return a.l < b.l;
15     return a.r > b.r;
16 }
17 void reset(){
18     for(int &b : BIT) b = 0;
19 }
20 void update(int p){
21     for(; p < 200004; p += p & -p){
22         BIT[p]++;
23     }
24 }
25 int query(int p){
26     int sum = 0;
27     for(; p > 0; p -= p & -p){
28         sum += BIT[p];
29     }
30     return sum;
31 }
32 signed main(){
33     int n, x, y, lst = 0, cnt = 0;
34     cin >> n;
35     for(int i = 0; i < n; i++){
36         cin >> x >> y;
37         X.pb({x, i});
38         Y.pb({y, i});
39         R[i].t = i;
40     }
41     sort(X.begin(), X.end());
42     sort(Y.begin(), Y.end());
43     for(pii p : X){
44         if(p.ff != lst) cnt++;
45         R[p.ss].l = cnt;
46         lst = p.ff;
47     }
48     lst = cnt = 0;
49     for(pii p : Y){
50         if(p.ff != lst) cnt++;
51         R[p.ss].r = cnt;
52         lst = p.ff;
53     }
54     sort(R.begin(), R.begin() + n, cmp);
55     for(int i = n - 1; i >= 0; i--){
56         ans[R[i].t] = query(R[i].r);
57         update(R[i].r);
58     }
59     for(int i = 0; i < n; i++){
60         cout << ans[i] << " ";
61         ans[i] = 0;
62     }
63     cout << "\n";
64     reset();
65     for(int i = 0; i < n; i++){
```

```

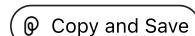
66         ans[R[i].t] = query(200001) - query(R[i].r - 1);
67         update(R[i].r);
68     }
69     for(int i = 0; i < n; i++){
70         cout << ans[i] << " ";
71     }
72     return 0;
73 }
```

Room Allocation (<https://cses.fi/problemset/task/1164>)

排序

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 struct day{
5     int t, c, p;
6 };
7 priority_queue<int, vector<int>, greater<int>> Q;
8 array<int, 200004> R;
9 vector<day> C;
10 bool cmp(day a, day b){
11     if(a.t != b.t) return a.t < b.t;
12     return a.c > b.c;
13 }
14 signed main(){
15     int n, l, r, now = 0, ans = 0;
16     cin >> n;
17     for(int i = 1; i <= n; i++){
18         cin >> l >> r;
19         C.pb({l, 1, i});
20         C.pb({r, -1, i});
21         Q.push(i);
22     }
23     sort(C.begin(), C.end(), cmp);
24     for(day d : C){
25         if(d.c > 0){
26             R[d.p] = Q.top();
27             Q.pop();
28         }else{
29             Q.push(R[d.p]);
30         }
31         now += d.c;
32         ans = max(ans, now);
33     }
34     cout << ans << "\n";
35     for(int i = 1; i <= n; i++){
36         cout << R[i] << " ";
37     }
38     return 0;
39 }
```

Factory Machines (<https://cses.fi/problemset/task/1620>)

二分搜

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> T;
5 int run(int n, int t){
6     int sum = 0;
7     for(int i = 0; i < n; i++){
8         sum += t / T[i];
9     }
10    return sum;
11 }
12 int BS(int n, int k){
13     int l = 0, r = 1e18 / n, mid;
14     r += r / n;
15     while(l != r){
16         mid = (l + r) / 2;
17         if(run(n, mid) < k) l = mid + 1;
18         else r = mid;
19     }
20    return l;
21 }
22 signed main(){
23     int n, k;
24     cin >> n >> k;
25     for(int i = 0; i < n; i++){
26         cin >> T[i];
27     }
28     cout << BS(n, k);
29     return 0;
30 }
```

Tasks and Deadlines (<https://cses.fi/problemset/task/1630>)

Greedy

```

1 #include <bits/stdc++.h>
2 #define int long long
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 array<pii, 200004> T;
8 signed main(){
9     int n, a, d, now = 0, ans = 0;
10    cin >> n;
11    for(int i = 0; i < n; i++){
12        cin >> a >> d;
13        T[i] = {a, d};
14    }
15    sort(T.begin(), T.begin() + n);
16    for(int i = 0; i < n; i++){
17        now += T[i].ff;
18        ans += T[i].ss - now;
19    }
20    cout << ans;
21    return 0;
22 }
```

Reading Books (<https://cses.fi/problemset/task/1631>)

數學

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 signed main(){
5     int n, sum = 0, lng = 0, t;
6     cin >> n;
7     for(int i = 0; i < n; i++){
8         cin >> t;
9         sum += t;
10        lng = max(lng, t);
11    }
12    if(lng * 2 < sum) cout << sum;
13    else cout << 2 * lng;
14    return 0;
15 }
```

Sum of Three Values (<https://cses.fi/problemset/task/1641>)

Map

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 5004> A;
4 map<int, int> M;
5 signed main(){
6     int n, x;
7     cin >> n >> x;
8     for(int i = 1; i <= n; i++){
9         cin >> A[i];
10    }
11    for(int i = 1; i <= n; i++){
12        for(int j = i + 1; j <= n; j++){
13            if(M.find(x - A[i] - A[j]) != M.end()){
14                cout << M[x - A[i] - A[j]] << " " << i << " " << j;
15                return 0;
16            }
17        }
18        M[A[i]] = i;
19    }
20    cout << "IMPOSSIBLE";
21    return 0;
22 }
```

Sum of Four Values (<https://cses.fi/problemset/task/1642>)

排序 雙指針

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 struct s{
4     int v, i, j;
5 };
6 array<int, 1004> A;
7 array<s, 100004> S;
8 bool cmp(s a, s b){
9     return a.v < b.v;
10 }
11 signed main(){
12     int n, x, cnt = 0, p;
13     cin >> n >> x;
14     for(int i = 1; i <= n; i++){
15         cin >> A[i];
16     }
17     for(int i = 1; i <= n; i++){
18         for(int j = i + 1; j <= n; j++){
19             S[cnt++] = {A[i] + A[j], i, j};
20         }
21     }
22     sort(S.begin(), S.begin() + cnt, cmp);
23     p = cnt - 1;
24     for(int i = 0; i < p; i++){
25         while(S[i].v + S[p].v > x) p--;
26         while(S[i].v + S[p].v == x && (S[i].i == S[p].i || S[i].i == S[p].j || S[i].j == S[p].i || S[i].j == S[p].j)){
27             if(S[i].v + S[p].v == x){
28                 cout << S[i].i << " " << S[i].j << " " << S[p].i << " " << S[p].j;
29                 return 0;
30             }
31         }
32     cout << "IMPOSSIBLE";
33     return 0;
34 }
```

Nearest Smaller Values (<https://cses.fi/problemset/task/1645>)

Monoton Stack

```

1 #include <bits/stdc++.h>
2 #define pii pair<int, int>
3 #define ff first
4 #define ss second
5 using namespace std;
6
7 signed main(){
8     int n, x;
9     stack<pii> S;
10    cin >> n;
11    S.push({0, 0});
12    for(int i = 1; i <= n; i++){
13        cin >> x;
14        while(x <= S.top().ff) S.pop();
15        cout << S.top().ss << " ";
16        S.push({x, i});
17    }
18    return 0;
19 }
```

Subarray Sums I (<https://cses.fi/problemset/task/1660>)

雙指針

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 200004> A;
4 signed main(){
5     int n, x, p = 0, sum = 0, ans = 0;
6     cin >> n >> x;
7     for(int i = 0; i < n; i++){
8         cin >> A[i];
9         sum += A[i];
10        while(sum > x){
11            sum -= A[p++];
12        }
13        if(sum == x) ans++;
14    }
15    cout << ans;
16    return 0;
17 }
```

Subarray Sums II (<https://cses.fi/problemset/task/1661>)

Map

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 map<int, int> M;
5 signed main(){
6     int n, x, a, sum = 0, ans = 0;
7     cin >> n >> x;
8     M[0]++;
9     for(int i = 0; i < n; i++){
10         cin >> a;
11         sum += a;
12         ans += M[sum - x];
13         M[sum]++;
14     }
15     cout << ans;
16     return 0;
17 }
```

Subarray Divisibility (<https://cses.fi/problemset/task/1662>)

暴力

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> mod;
5 signed main(){
6     int n, a, sum = 0, ans = 0;
7     cin >> n;
8     mod[0]++;
9     for(int i = 0; i < n; i++){
10         cin >> a;
11         sum += a;
12         mod[((sum % n) + n) % n]++;
13     }
14     for(int i = 0; i < n; i++){
15         ans += (mod[i] * (mod[i] - 1)) / 2;
16     }
17     cout << ans;
18     return 0;
19 }
```

Subarray Distinct Values (<https://cses.fi/problemset/task/2428>)

雙指針

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> X;
5 map<int, int> M;
6 signed main(){
7     int n, k, p = 0, now = 0, ans = 0;
8     cin >> n >> k;
9     for(int i = 0; i < n; i++){
10         cin >> X[i];
11         if(M[X[i]] == 0) now++;
12         M[X[i]]++;
13         while(now > k){
14             M[X[p]]--;
15             if(M[X[p]] == 0) now--;
16             p++;
17         }
18         ans += i - p + 1;
19     }
20     cout << ans;
21     return 0;
22 }
```

Array Division (<https://cses.fi/problemset/task/1085>)

二分搜

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<int, 200004> X;
5 int cost(int n, int x){
6     int k = 1, sum = 0;
7     for(int i = 0; i < n; i++){
8         if(X[i] > x) return 1e9;
9         if(sum + X[i] > x){
10             k++;
11             sum = 0;
12         }
13         sum += X[i];
14     }
15     return k;
16 }
17 int BS(int n, int k){
18     int l = 0, r = 2e14, mid;
19     while(l != r){
20         mid = (l + r) / 2;
21         if(cost(n, mid) > k) l = mid + 1;
22         else r = mid;
23     }
24     return l;
25 }
26 signed main(){
27     int n, k;
28     cin >> n >> k;
29     for(int i = 0; i < n; i++){
30         cin >> X[i];
31     }
32     cout << BS(n, k);
33     return 0;
34 }
```

Sliding Median (<https://cses.fi/problemset/task/1076>)

排序 Map BIT 雙指針

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 map<int, int> M;
4 array<int, 200004> X, L, BIT, m;
5 void update(int p, int x){
6     for(; p < 200004; p += p & -p){
7         BIT[p] += x;
8     }
9 }
10 int find(int x){
11     int sum = 0, p = 0;
12     for(int i = (1 << 17); i > 0; i >= 1){
13         if(p + i < 200004 && sum + BIT[p + i] < x){
14             p += i;
15             sum += BIT[p];
16         }
17     }
18     return p + 1;
19 }
20 signed main(){
21     int n, k, cnt = 0, lst = 0;
22     cin >> n >> k;
23     for(int i = 0; i < n; i++){
24         cin >> X[i];
25         L[i] = X[i];
26     }
27     sort(L.begin(), L.begin() + n);
28     for(int i = 0; i < n; i++){
29         if(L[i] != lst) cnt++;
30         M[L[i]] = cnt;
31         m[cnt] = L[i];
32         lst = L[i];
33     }
34     for(int i = 0; i < k; i++){
35         update(M[X[i]], 1);
36     }
37     cout << m[find(k / 2 + (k % 2))] << " ";
38     for(int i = k; i < n; i++){
39         update(M[X[i]], 1);
40         update(M[X[i - k]], -1);
41         cout << m[find(k / 2 + (k % 2))] << " ";
42     }
43     return 0;
44 }
```

Sliding Cost (<https://cses.fi/problemset/task/1077>)

排序 Map BIT 雙指針

```
1 #include <bits/stdc++.h>
2 #define pii pair<int, int>
3 #define ff first
4 #define ss second
5 #define int long long
6 using namespace std;
7 map<int, int> M;
8 array<int, 200004> X;
9 array<pii, 200004> L;
10 array<int, 200004> B, C;
11 void update(int p, int x){
12     for(; p < 200004; p += p & -p){
13         B[p] += x / abs(x);
14         C[p] += x;
15     }
16 }
17 int query(int p){
18     int sum = 0;
19     for(; p > 0; p -= p & -p){
20         sum += C[p];
21     }
22     return sum;
23 }
24 int find(int x){
25     int sum = 0, p = 0;
26     for(int i = (1 << 17); i > 0; i >= 1){
27         if(p + i < 200004 && sum + B[p + i] < x){
28             p += i;
29             sum += B[p];
30         }
31     }
32     return p + 1;
33 }
34 signed main(){
35     int n, k, cnt = 0, p;
36     cin >> n >> k;
37     for(int i = 0; i < n; i++){
38         cin >> X[i];
39         L[i] = {X[i], i};
40     }
41     sort(L.begin(), L.begin() + n);
42     for(int i = 0; i < n; i++){
43         cnt++;
44         M[L[i].ss] = cnt;
45     }
46     for(int i = 0; i < k; i++){
47         update(M[i], X[i]);
48     }
49     p = find(k / 2 + (k % 2));
50     if(k & 1) cout << query(200001) - query(p) - query(p - 1) << " ";
51     else cout << query(200001) - 2 * query(p) << " ";
52     for(int i = k; i < n; i++){
53         update(M[i], X[i]);
54         update(M[i - k], -X[i - k]);
55         p = find(k / 2 + (k % 2));
56         if(k & 1) cout << query(200001) - query(p) - query(p - 1) << " ";
57         else cout << query(200001) - 2 * query(p) << " ";
58     }
59     return 0;
60 }
```

Movie Festival II (<https://cses.fi/problemset/task/1632>)

Set 排序

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 bool cmp(pii a, pii b){
8     if(a.ss != b.ss) return a.ss < b.ss;
9     return a.ff < b.ff;
10 }
11 vector<pii> M;
12 multiset<int> F;
13 signed main(){
14     int n, k, ans = 0, l, r, siz = 0;
15     cin >> n >> k;
16     for(int i = 0; i < n; i++){
17         cin >> l >> r;
18         M.pb({l, r});
19     }
20     sort(M.begin(), M.end(), cmp);
21     for(int i = 0; i < k; i++){
22         F.insert(0);
23     }
24     for(pii m : M){
25         auto it = F.upper_bound(m.ff);
26         if(it == F.begin()) continue;
27         F.erase(--it);
28         F.insert(m.ss);
29         ans++;
30     }
31     cout << ans;
32     return 0;
33 }
```

Maximum Subarray Sum II (<https://cses.fi/problemset/task/1644>)

Monoton Deque

```

1 #include <bits/stdc++.h>
2 #define int long long
3 #define pb push_back
4 #define pf push_front
5 #define ppb pop_back
6 #define ppf pop_front
7 #define pii pair<int, int>
8 #define ff first
9 #define ss second
10 using namespace std;
11 deque<pii> Q;
12 array<int, 200004> S;
13 signed main(){
14     int n, a, b, x, ans = -1e18;
15     cin >> n >> a >> b;
16     for(int i = 1; i <= n; i++){
17         cin >> x;
18         S[i] = x + S[i - 1];
19         if(!Q.empty() && Q.front().ss < i - b) Q.ppf();
20         while(i >= a && !Q.empty() && S[i - a] < Q.back().ff) Q.ppb();
21         if(i >= a) Q.pb({S[i - a], i - a});
22         if(!Q.empty()) ans = max(ans, S[i] - Q.front().ff);
23     }
24     cout << ans;
25     return 0;
26 }
```

Dynamic Programming

Dice Combinations (<https://cses.fi/problemset/task/1633>)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<int, 1000004> dp;
5 int DP(int n){
6     dp[0] = 1;
7     for(int i = 1; i <= n; i++){
8         for(int j = 1; j <= min(i, 6); j++){
9             dp[i] += dp[i - j];
10            dp[i] %= mod;
11        }
12    }
13    return dp[n];
14 }
15 signed main(){
16     int n;
17     cin >> n;
18     cout << DP(n);
19     return 0;
20 }
```

Minimizing Coins (<https://cses.fi/problemset/task/1634>)

DP

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 const int inf = 1e6 + 1;
4 array<int, 104> C;
5 array<int, 1000004> dp;
6 int DP(int x){
7     for(int i = 1; i <= x; i++){
8         dp[i] = inf;
9         for(int c : C){
10             if(c > i) continue;
11             dp[i] = min(dp[i], dp[i - c] + 1);
12         }
13     }
14     return dp[x] == inf? -1 : dp[x];
15 }
16 signed main(){
17     int n, x;
18     cin >> n >> x;
19     for(int i = 0; i < n; i++){
20         cin >> C[i];
21     }
22     cout << DP(x);
23     return 0;
24 }
```

Coin Combinations I (<https://cses.fi/problemset/task/1635>)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<int, 104> C;
5 array<int, 1000004> dp;
6 int DP(int x){
7     dp[0] = 1;
8     for(int i = 1; i <= x; i++){
9         for(int c : C){
10             if(c > i || !c) continue;
11             dp[i] += dp[i - c];
12             dp[i] %= mod;
13         }
14     }
15     return dp[x];
16 }
17 signed main(){
18     int n, x;
19     cin >> n >> x;
20     for(int i = 0; i < n; i++){
21         cin >> C[i];
22     }
23     cout << DP(x);
24     return 0;
25 }
```

Coin Combinations II (<https://cses.fi/problemset/task/1636>)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<int, 104> C;
5 array<int, 1000004> dp;
6 int DP(int x){
7     dp[0] = 1;
8     for(int c : C){
9         if(!c) break;
10        for(int i = 1; i <= x; i++){
11            if(i < c) continue;
12            dp[i] += dp[i - c];
13            dp[i] %= mod;
14        }
15    }
16    return dp[x];
17 }
18 signed main(){
19     int n, x;
20     cin >> n >> x;
21     for(int i = 0; i < n; i++){
22         cin >> C[i];
23     }
24     cout << DP(x);
25     return 0;
26 }
```

[Removing Digits](https://cses.fi/problemset/task/1637)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int inf = 1e6;
4 array<int, 1000004> dp;
5 int DP(int n){
6     int t;
7     for(int i = 1; i <= n; i++){
8         dp[i] = inf;
9         t = i;
10        while(t > 0){
11            dp[i] = min(dp[i], dp[i - t % 10] + 1);
12            t /= 10;
13        }
14    }
15    return dp[n];
16}
17 signed main(){
18     int n;
19     cin >> n;
20     cout << DP(n);
21     return 0;
22 }
```

[Grid Paths](https://cses.fi/problemset/task/1638)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<array<char, 1004>, 1004> G;
5 array<array<int, 1004>, 1004> dp;
6 int DP(int i, int j){
7     if(dp[i][j]) return dp[i][j];
8     if(!i || !j || G[i][j] == '*') return 0;
9     G[i][j] = '*';
10    return dp[i][j] = (DP(i, j - 1) + DP(i - 1, j)) % mod;
11}
12 signed main(){
13     int n;
14     cin >> n;
15     for(int i = 1; i <= n; i++){
16         for(int j = 1; j <= n; j++){
17             cin >> G[i][j];
18         }
19     }
20    dp[1][1] = G[1][1] == '*' ? 0 : 1;
21    cout << DP(n, n);
22    return 0;
23 }
```

Book Shop (<https://cses.fi/problemset/task/1158>)

背包

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 1004> H, S;
4 array<int, 100004> dp;
5 int bag(int n, int x){
6     int ans = 0;
7     for(int j = 0; j < n; j++){
8         for(int i = x; i >= H[j]; i--){
9             dp[i] = max(dp[i], dp[i - H[j]] + S[j]);
10            ans = max(ans, dp[i]);
11        }
12    }
13    return ans;
14 }
15 signed main(){
16     int n, x;
17     cin >> n >> x;
18     for(int i = 0; i < n; i++){
19         cin >> H[i];
20     }
21     for(int i = 0; i < n; i++){
22         cin >> S[i];
23     }
24     cout << bag(n, x);
25     return 0;
26 }
```

Array Description (<https://cses.fi/problemset/task/1746>)

DP

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<int, 100004> X;
5 array<array<int, 104>, 100004> dp;
6 int DP(int n, int m){
7     int ans = 0;
8     for(int i = 2; i <= n; i++){
9         if(X[i]){
10             for(int k = -1; k <= 1; k++){
11                 dp[i][X[i]] += dp[i - 1][X[i] + k];
12                 dp[i][X[i]] %= mod;
13             }
14         }else{
15             for(int j = 1; j <= m; j++){
16                 for(int k = -1; k <= 1; k++){
17                     dp[i][j] += dp[i - 1][j + k];
18                     dp[i][j] %= mod;
19                 }
20             }
21         }
22     }
23     for(int i = 1; i <= m; i++){
24         ans += dp[n][i];
25         ans %= mod;
26     }
27     return ans;
28 }
29 signed main(){
30     int n, m;
31     cin >> n >> m;
32     for(int i = 1; i <= n; i++){
33         cin >> X[i];
34     }
35     if(X[1]) dp[1][X[1]] = 1;
36     else{
37         for(int i = 1; i <= m; i++){
38             dp[1][i] = 1;
39         }
40     }
41     cout << DP(n, m);
42     return 0;
43 }
```

Counting Towers (<https://cses.fi/problemset/task/2413>)

DP

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 const int mod = 1e9 + 7;
5 array<array<int, 8>, 1000004> dp;
6 void DP(){
7     /*
8     _|_ = 0, 2, 4
9     */
10    dp[0][7] = dp[0][5] = 1;
11    for(int i = 1; i <= 1e6; i++){
12        dp[i][0] = (dp[i - 1][0] + dp[i - 1][5]) % mod;
13        dp[i][2] = (dp[i - 1][7] + dp[i - 1][3] + dp[i - 1][2] + dp[i - 1][6]);
14        dp[i][3] = (dp[i - 1][7] + dp[i - 1][3] + dp[i - 1][2] + dp[i - 1][6]);
15        dp[i][5] = (dp[i - 1][7] + dp[i - 1][3] + dp[i - 1][2] + dp[i - 1][6]);
16        dp[i][6] = (dp[i - 1][7] + dp[i - 1][3] + dp[i - 1][2] + dp[i - 1][6]);
17        dp[i][7] = (dp[i - 1][7] + dp[i - 1][3] + dp[i - 1][2] + dp[i - 1][6]);
18    }
19 }
20 signed main(){
21     int t, n;
22     cin >> t;
23     DP();
24     while(t--){
25         cin >> n;
26         cout << dp[n][5] << "\n";
27     }
28     return 0;
29 }
```

Edit Distance (<https://cses.fi/problemset/task/1639>)

DP

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 string A, B;
4 array<array<bool, 5004>, 5004> vis;
5 array<array<int, 5004>, 5004> dp;
6 int ED(int i, int j){
7     if(!i || !j) return max(i, j);
8     if(vis[i][j]) return dp[i][j];
9     vis[i][j] = 1;
10    if(A[i - 1] == B[j - 1]) return dp[i][j] = ED(i - 1, j - 1);
11    return dp[i][j] = min({ED(i - 1, j - 1), ED(i - 1, j), ED(i, j - 1)}) + 1;
12 }
13 signed main(){
14     cin >> A >> B;
15     cout << ED(A.size(), B.size());
16     return 0;
17 }
```

Rectangle Cutting (<https://cses.fi/problemset/task/1744>)

DP

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 const int inf = 25e4;
4 array<array<int, 504>, 504> dp;
5 int DP(int a, int b){
6     for(int i = 1; i <= a; i++){
7         for(int j = 1; j <= b; j++){
8             if(i == j) continue;
9             dp[i][j] = inf;
10            for(int k = 1; k < i; k++){
11                dp[i][j] = min(dp[i][j], dp[k][j] + dp[i - k][j] + 1);
12            }
13            for(int k = 1; k < j; k++){
14                dp[i][j] = min(dp[i][j], dp[i][k] + dp[i][j - k] + 1);
15            }
16        }
17    }
18    return dp[a][b];
19 }
20 signed main(){
21     int a, b;
22     cin >> a >> b;
23     cout << DP(a, b);
24     return 0;
25 }
```

Money Sums (<https://cses.fi/problemset/task/1745>)

DP

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 104> X;
4 array<bool, 100004> dp;
5 void DP(int n){
6     int cnt = 0;
7     dp[0] = 1;
8     for(int x : X){
9         for(int i = n; i >= x; i--){
10             dp[i] |= dp[i - x];
11         }
12     }
13     for(int i = 1; i <= n; i++){
14         cnt += dp[i];
15     }
16     cout << cnt << "\n";
17     for(int i = 1; i <= n; i++){
18         if(dp[i]) cout << i << " ";
19     }
20 }
21 signed main(){
22     int n, sum = 0;
23     cin >> n;
24     for(int i = 0; i < n; i++){
25         cin >> X[i];
26         sum += X[i];
27     }
28     DP(sum);
29     return 0;
30 }
```

Removal Game (<https://cses.fi/problemset/task/1097>)

DP 賽局

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 struct pii{
5     int ff, ss;
6     pii(){ff = ss = 0;};
7     pii(int f, int s): ff(f), ss(s){}
8     pii operator+(pii p){
9         return pii(ff + p.ff, ss + p.ss);
10    }
11 };
12 array<int, 5004> X;
13 array<array<pii, 5004>, 5004> dp;
14 pii maxf(pii a, pii b){
15     if(a.ff > b.ff) return a;
16     return b;
17 }
18 pii maxs(pii a, pii b){
19     if(a.ss > b.ss) return a;
20     return b;
21 }
22 pii play(int i, int j, int p){
23     if(i > j) return {0, 0};
24     if(dp[i][j].ff && dp[i][j].ss) return dp[i][j];
25     if(!p){
26         return dp[i][j] = maxf(play(i + 1, j, 1) + pii(X[i], 0), play(i, j - 1
27     }else{
28         return dp[i][j] = maxs(play(i + 1, j, 0) + pii(0, X[i]), play(i, j - 1
29     }
30 }
31 signed main(){
32     int n;
33     cin >> n;
34     for(int i = 1; i <= n; i++){
35         cin >> X[i];
36     }
37     cout << play(1, n, 0).ff;
38     return 0;
39 }
```

Two Sets II (<https://cses.fi/problemset/task/1093>)

DP 模逆元

```

1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 const int mod = 1e9 + 7;
5 array<int, 150004> dp;
6 int DP(int n, int x){
7     dp[0] = 1;
8     for(int j = 1; j <= n; j++){
9         for(int i = x; i >= j; i--){
10            dp[i] += dp[i - j];
11            dp[i] %= mod;
12        }
13    }
14    return (dp[x] * (int)(5e8 + 4)) % mod;
15}
16 signed main(){
17     int n, sum = 0;
18     cin >> n;
19     for(int i = 1; i <= n; i++){
20         sum += i;
21     }
22     if(sum & 1) cout << 0;
23     else cout << DP(n, sum / 2);
24     return 0;
25 }
```

Increasing Subsequence (<https://cses.fi/problemset/task/1145>)

LIS

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 200004> X;
5 int LIS(int n){
6     vector<int> lis;
7     for(int i = 0; i < n ;i++){
8         auto it = lower_bound(lis.begin(), lis.end(), X[i]);
9         if(it == lis.end()) lis.pb(X[i]);
10        else *it = X[i];
11    }
12    return lis.size();
13}
14 signed main(){
15     int n;
16     cin >> n;
17     for(int i = 0; i < n; i++){
18         cin >> X[i];
19     }
20     cout << LIS(n);
21     return 0;
22 }
```

Projects (<https://cses.fi/problemset/task/1140>)

排序 DP

```
1 #include <bits/stdc++.h>
2 #define int long long
3 #define pb push_back
4 using namespace std;
5 struct w{
6     int l, r, p;
7 };
8 bool cmpl(w a, w b){
9     return a.l < b.l;
10 }
11 bool cmpr(w a, w b){
12     return a.r < b.r;
13 }
14 array<int, 400004> dp;
15 array<w, 200004> W;
16 vector<w> L;
17 signed main(){
18     int n, l, r, p, cnt = 0, lst = 0, ans = 0;
19     cin >> n;
20     for(int i = 0; i < n; i++){
21         cin >> l >> r >> p;
22         W[i].p = p;
23         L.pb({l, i, 0});
24         L.pb({r, i, 1});
25     }
26     sort(L.begin(), L.end(), cmpl);
27     for(w ll : L){
28         if(ll.l != lst) cnt++;
29         if(ll.p) W[ll.r].r = cnt;
30         else W[ll.r].l = cnt;
31         lst = ll.l;
32     }
33     sort(W.begin(), W.begin() + n, cmpr);
34     p = 0;
35     for(int i = 1; i <= 2 * n; i++){
36         dp[i] = dp[i - 1];
37         while(p < n && W[p].r == i){
38             dp[i] = max(dp[i], dp[W[p].l - 1] + W[p].p);
39             p++;
40         }
41         ans = max(ans, dp[i]);
42     }
43     cout << ans;
44     return 0;
45 }
```

Elevator Rides (<https://cses.fi/problemset/task/1653>)

狀態DP

```
1 #include <bits/stdc++.h>
2 #define pii pair<int, int>
3 #define ff first
4 #define ss second
5 using namespace std;
6 array<int, 24> W;
7 array<pii, 1 << 20> dp;
8 int DP(int n, int x){
9     int w, t;
10    dp[0] = {1, 0};
11    for(int i = 1; i < (1 << n); i++){
12        dp[i] = {24, 0};
13        for(int j = 0; j < n; j++){
14            if(i & (1 << j)){
15                t = dp[i ^ (1 << j)].ff;
16                w = dp[i ^ (1 << j)].ss;
17                if(w + W[j] > x){
18                    t++;
19                    w = min(W[j], w);
20                }else w += W[j];
21                dp[i] = min(dp[i], {t, w});
22            }
23        }
24    }
25    return dp[(1 << n) - 1].ff;
26}
27 signed main(){
28     int n, x;
29     cin >> n >> x;
30     for(int i = 0; i < n; i++){
31         cin >> W[i];
32     }
33     cout << DP(n, x);
34     return 0;
35 }
```

Counting Tilings (<https://cses.fi/problemset/task/2181>)

輪廓DP

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 const int mod = 1e9 + 7;
4 array<array<int, 1 << 12>, 2> dp;
5 void cpy(int n){
6     for(int i = 0; i < 1 << n; i++){
7         dp[0][i] = dp[1][i];
8         dp[1][i] = 0;
9     }
10 }
11 int DP(int n, int m){
12     dp[1][0] = 1;
13     for (int i = 0; i < m; i++) {
14         for (int j = 0; j < n; j++) {
15             cpy(n);
16             for (int s = 0; s < 1 << n; s++){
17                 if (dp[0][s]) {
18                     if (j != n - 1 && !(s >> j & 3)){
19                         dp[1][s ^ 1 << j + 1] += dp[0][s];
20                         dp[1][s ^ 1 << j + 1] %= mod;
21                     }
22                     dp[1][s ^ 1 << j] += dp[0][s];
23                     dp[1][s ^ 1 << j] %= mod;
24                 }
25             }
26         }
27     }
28     return dp[1][0];
29 }
30 signed main(){
31     int n, m;
32     cin >> n >> m;
33     cout << DP(n, m);
34     return 0;
35 }
```

Counting Numbers (<https://cses.fi/problemset/task/2220>)

數位DP

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 vector<int> dig;
6 array<array<array<array<int, 2>, 2>, 10>, 20> dp;
7 int dfs(int d, int pre, bool t, bool z/*tight, zero*/){
8     if(d < 0){
9         return 1;
10    }
11    if(dp[d][pre][t][z]) return dp[d][pre][t][z];
12    int sum = 0;
13    if(t){
14        if(z || pre) sum += dfs(d - 1, 0, !dig[d], z);
15        for(int i = 1; i <= dig[d]; i++){
16            if(i != pre) sum += dfs(d - 1, i, i == dig[d], 0);
17        }
18    }else if(z){
19        sum += dfs(d - 1, 0, 0, 1);
20        for(int i = 1; i <= 9; i++){
21            sum += dfs(d - 1, i, 0, 0);
22        }
23    }else{
24        for(int i = 0; i <= 9; i++){
25            if(i != pre) sum += dfs(d - 1, i, 0, 0);
26        }
27    }
28    return dp[d][pre][t][z] = sum;
29}
30 int DP(int x){
31     if(!x) return 1;
32     for(int i = 0; i < 20; i++){
33         for(int j = 0; j < 10; j++){
34             for(int k = 0; k < 2; k++){
35                 for(int l = 0; l < 2; l++){
36                     dp[i][j][k][l] = 0;
37                 }
38             }
39         }
40     }
41     dig.clear();
42     while(x > 0){
43         dig.pb(x % 10);
44         x /= 10;
45     }
46     return dfs(dig.size() - 1, 0, 1, 1);
47 }
48 signed main(){
49     int a, b;
50     cin >> a >> b;
51     cout << DP(b) - (a? DP(a - 1) : 0);
52     return 0;
53 }
```

Graph Algorithms

Counting Rooms (<https://cses.fi/problemset/task/1192>)

DFS

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 int n, m;
4 array<int, 4> dx = {0, -1, 0, 1}, dy = {1, 0, -1, 0};
5 array<array<char, 1004>, 1004> G;
6 void dfs(int i, int j){
7     if(!i || !j || i > n || j > m || G[i][j] == '#') return;
8     G[i][j] = '#';
9     for(int k = 0; k < 4; k++){
10         dfs(i + dx[k], j + dy[k]);
11     }
12 }
13 signed main(){
14     int ans = 0;
15     cin >> n >> m;
16     for(int i = 1; i <= n; i++){
17         for(int j = 1; j <= m; j++){
18             cin >> G[i][j];
19         }
20     }
21     for(int i = 1; i <= n; i++){
22         for(int j = 1; j <= m; j++){
23             if(G[i][j] == '.'){
24                 dfs(i, j);
25                 ans++;
26             }
27         }
28     }
29     cout << ans;
30     return 0;
31 }
```

Labyrinth (<https://cses.fi/problemset/task/1193>)

BFS

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 struct pos{
4     int x, y, c;
5 };
6 int n, m;
7 array<int, 4> dx = {0, 1, 0, -1}, dy = {1, 0, -1, 0};
8 array<array<char, 1004>, 1004> G;
9 array<array<pos, 1004>, 1004> L;
10 pos bfs(pos s){
11     pos u;
12     queue<pos> Q;
13     Q.push(s);
14     while(!Q.empty()){
15         u = Q.front();
16         Q.pop();
17         if(!u.x || !u.y || u.x > n || u.y > m || G[u.x][u.y] == '#') continue;
18         if(G[u.x][u.y] == 'B') return u;
19         G[u.x][u.y] = '#';
20         L[u.x][u.y] = u;
21         for(int i = 0; i < 4; i++){
22             Q.push({u.x + dx[i], u.y + dy[i], i});
23         }
24     }
25     return {0, 0, -1};
26 }
27 void print(pos now, int cnt){
28     if(now.c == -1){
29         if(now.x) cout << "YES\n" << cnt << "\n";
30         else cout << "NO\n";
31         return;
32     }
33     print(L[now.x - dx[now.c]][now.y - dy[now.c]], cnt + 1);
34     if(now.c == 0) cout << "R";
35     else if(now.c == 1) cout << "D";
36     else if(now.c == 2) cout << "L";
37     else cout << "U";
38 }
39 signed main(){
40     pos s;
41     cin >> n >> m;
42     for(int i = 1; i <= n; i++){
43         for(int j = 1; j <= m; j++){
44             cin >> G[i][j];
45             if(G[i][j] == 'A'){
46                 s = {i, j, -1};
47             }
48         }
49     }
50     print(bfs(s), 0);
51     return 0;
52 }
```

Building Roads (<https://cses.fi/problemset/task/1666>)

DSU

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 array<int, 100004> DSU;
8 vector<pii> R;
9 int find(int x){
10     if(DSU[x] == x) return x;
11     return DSU[x] = find(DSU[x]);
12 }
13 void onion(int a, int b){
14     int A = find(a), B = find(b);
15     DSU[B] = A;
16 }
17 signed main(){
18     int n, m, a, b, cnt = 0;
19     cin >> n >> m;
20     for(int i = 1; i <= n; i++){
21         DSU[i] = i;
22     }
23     while(m--){
24         cin >> a >> b;
25         onion(a, b);
26     }
27     for(int i = 2; i <= n; i++){
28         if(find(1) != find(i)){
29             cnt++;
30             R.pb({1, i});
31             onion(1, i);
32         }
33     }
34     cout << cnt << "\n";
35     for(pii r : R){
36         cout << r.ff << " " << r.ss << "\n";
37     }
38     return 0;
39 }
```

Message Route (<https://cses.fi/problemset/task/1667>)

BFS

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 100004> L;
5 array<bool, 100004> vis;
6 array<vector<int>, 100004> G;
7 bool bfs(int s, int t){
8     int u;
9     queue<int> Q;
10    Q.push(s);
11    while(!Q.empty()){
12        int u = Q.front();
13        Q.pop();
14        if(vis[u]) continue;
15        vis[u] = 1;
16        if(u == t) return 1;
17        for(int v : G[u]){
18            if(L[v]) continue;
19            L[v] = u;
20            Q.push(v);
21        }
22    }
23    return 0;
24 }
25 void print(int u, int cnt){
26     if(u == 1){
27         cout << cnt << "\n1 ";
28         return;
29     }
30     print(L[u], cnt + 1);
31     cout << u << " ";
32 }
33 signed main(){
34     int n, m, a, b;
35     cin >> n >> m;
36     while(m--){
37         cin >> a >> b;
38         G[a].pb(b);
39         G[b].pb(a);
40     }
41     if(bfs(1, n)) print(n, 1);
42     else cout << "IMPOSSIBLE";
43     return 0;
44 }
```

[Building Teams](https://cses.fi/problemset/task/1668)

DFS

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<vector<int>, 100004> G;
5 array<int, 100004> team;
6 bool dfs(int u, int t){
7     team[u] = t;
8     bool ok = 1;
9     for(int v : G[u]){
10         if(team[v]){
11             if(team[v] == t) return 0;
12             continue;
13         }
14         ok &= dfs(v, 3 - t);
15     }
16     return ok;
17 }
18 signed main(){
19     int n, m, a, b;
20     bool ans = 1;
21     cin >> n >> m;
22     while(m--){
23         cin >> a >> b;
24         G[a].pb(b);
25         G[b].pb(a);
26     }
27     for(int i = 1; i <= n; i++){
28         if(!team[i]) ans &= dfs(i, 1);
29     }
30     if(ans){
31         for(int i = 1; i <= n; i++){
32             cout << team[i] << " ";
33         }
34     }else cout << "IMPOSSIBLE";
35     return 0;
36 }
```

Round Trip (<https://cses.fi/problemset/task/1669>)

DFS

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<vector<int>, 100004> G;
5 array<int, 100004> vis;
6 vector<int> ans;
7 int s;
8 bool dfs(int u, int p, int pre){
9     vis[u] = p;
10    for(int v : G[u]){
11        if(v == pre) continue;
12        if(vis[v]){
13            if(vis[v] == p){
14                s = v;
15                ans.pb(v);
16                ans.pb(u);
17                return 1;
18            }
19            continue;
20        }
21        if(dfs(v, p, u)){
22            ans.pb(u);
23            if(s == u){
24                cout << ans.size() << "\n";
25                for(int a : ans){
26                    cout << a << " ";
27                }
28                exit(0);
29            }
30            return 1;
31        }
32    }
33    return 0;
34 }
35 signed main(){
36     int n, m, a, b;
37     cin >> n >> m;
38     while(m--){
39         cin >> a >> b;
40         G[a].pb(b);
41         G[b].pb(a);
42     }
43     for(int i = 1; i <= n; i++){
44         if(!vis[i]){
45             ans.clear();
46             dfs(i, i, 0);
47         }
48     }
49     cout << "IMPOSSIBLE";
50     return 0;
51 }
```

Monsters (<https://cses.fi/problemset/task/1194>)

BFS


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define top front
4 #define pii pair<int, int>
5 #define x first
6 #define y second
7 using namespace std;
8 int n, m;
9 array<int, 4> dx = {0, 1, 0, -1}, dy = {1, 0, -1, 0};
10 array<array<char, 1004>, 1004> G;
11 array<array<int, 1004>, 1004> disa, dism, L;
12 vector<pii> M;
13 pii s;
14 void bfsa(){
15     pii u;
16     int nx, ny;
17     queue<pii> Q;
18     Q.push(s);
19     disa[s.x][s.y] = 0;
20     while(!Q.empty()){
21         u = Q.top();
22         Q.pop();
23         if(!u.x || !u.y || u.x > n || u.y > m) continue;
24         for(int i = 0; i < 4; i++){
25             nx = u.x + dx[i];
26             ny = u.y + dy[i];
27             if(disa[nx][ny] >= 0 || G[nx][ny] == '#') continue;
28             disa[nx][ny] = disa[u.x][u.y] + 1;
29             L[nx][ny] = i;
30             Q.push({nx, ny});
31         }
32     }
33 }
34 void bfsm(){
35     pii u;
36     int nx, ny;
37     queue<pii> Q;
38     for(pii p : M){
39         dism[p.x][p.y] = 0;
40         Q.push(p);
41     }
42     while(!Q.empty()){
43         u = Q.top();
44         Q.pop();
45         if(!u.x || !u.y || u.x > n || u.y > m) continue;
46         for(int i = 0; i < 4; i++){
47             nx = u.x + dx[i];
48             ny = u.y + dy[i];
49             if(dism[nx][ny] >= 0 || G[nx][ny] == '#') continue;
50             dism[nx][ny] = dism[u.x][u.y] + 1;
51             Q.push({nx, ny});
52         }
53     }
54 }
55 void print(pii now){
56     if(now == s) return;
57     int l = L[now.x][now.y];
58     print({now.x - dx[l], now.y - dy[l]});
59     if(l == 0) cout << "R";
60     else if(l == 1) cout << "D";
61     else if(l == 2) cout << "L";
62     else cout << "U";
63 }
64 signed main(){
65     cin >> n >> m;
```

```

66     for(int i = 1; i <= n; i++){
67         for(int j = 1; j <= m; j++){
68             cin >> G[i][j];
69             if(G[i][j] == 'M') M.pb({i, j});
70             if(G[i][j] == 'A') s = {i, j};
71             disa[i][j] = dism[i][j] = -1;
72         }
73     }
74     bfsa();
75     bfsm();
76     for(int i = 1; i <= m; i++){
77         if(disa[1][i] >= 0 && (dism[1][i] < 0 || disa[1][i] < dism[1][i])){
78             cout << "YES\n";
79             cout << disa[1][i] << "\n";
80             print({1, i});
81             return 0;
82         }
83         if(disa[n][i] >= 0 && (dism[n][i] < 0 || disa[n][i] < dism[n][i])){
84             cout << "YES\n";
85             cout << disa[n][i] << "\n";
86             print({n, i});
87             return 0;
88         }
89     }
90     for(int i = 1; i <= n; i++){
91         if(disa[i][1] >= 0 && (dism[i][1] < 0 || disa[i][1] < dism[i][1])){
92             cout << "YES\n";
93             cout << disa[i][1] << "\n";
94             print({i, 1});
95             return 0;
96         }
97         if(disa[i][m] >= 0 && (dism[i][m] < 0 || disa[i][m] < dism[i][m])){
98             cout << "YES\n";
99             cout << disa[i][m] << "\n";
100            print({i, m});
101            return 0;
102        }
103    }
104    cout << "NO";
105    return 0;
106}

```

Shortest Routes I (<https://cses.fi/problemset/task/1671>)

Dijkstra

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int v, w;
7     E(){}
8     E(int v, int w): w(w), v(v){}
9     E operator+(E e){
10         return E(v, w + e.w);
11     }
12 };
13 struct cmp{
14     bool operator()(E a, E b){
15         return a.w > b.w;
16     }
17 };
18 array<vector<E>, 100004> G;
19 array<int, 100004> dis;
20 void run(int s){
21     E u;
22     priority_queue<E, vector<E>, cmp> Q;
23     Q.push(E(s, 1));
24     while(!Q.empty()){
25         u = Q.top();
26         Q.pop();
27         if(dis[u.v]) continue;
28         dis[u.v] = u.w;
29         for(E e : G[u.v]){
30             Q.push(e + u);
31         }
32     }
33 }
34 signed main(){
35     int n, m, a, b, c;
36     cin >> n >> m;
37     while(m--){
38         cin >> a >> b >> c;
39         G[a].pb(E(b, c));
40     }
41     run(1);
42     for(int i = 1; i <= n; i++){
43         cout << dis[i] - 1 << " ";
44     }
45     return 0;
46 }
```

Shortest Routes II (<https://cses.fi/problemset/task/1672>)

Floyd Warshall

```
1 #include <bits/stdc++.h>
2 #define int long long
3 using namespace std;
4 array<array<int, 504>, 504> dis;
5 int min(int a, int b){
6     if(a < 0) return b;
7     if(b < 0) return a;
8     return a < b? a : b;
9 }
10 int add(int a, int b){
11     if(a < 0 || b < 0) return -1;
12     return a + b;
13 }
14 void run(int n){
15     for(int k = 1; k <= n; k++){
16         for(int i = 1; i <= n; i++){
17             for(int j = 1; j <= n; j++){
18                 if(i == j) continue;
19                 dis[i][j] = min(dis[i][j], add(dis[i][k], dis[k][j]));
20             }
21         }
22     }
23 }
24 signed main(){
25     int n, m, q, a, b, c;
26     cin >> n >> m >> q;
27     for(int i = 1; i <= n; i++){
28         for(int j = 1; j <= n; j++){
29             dis[i][j] = -1;
30         }
31         dis[i][i] = 0;
32     }
33     while(m--){
34         cin >> a >> b >> c;
35         dis[a][b] = dis[b][a] = min(dis[a][b], c);
36     }
37     run(n);
38     while(q--){
39         cin >> a >> b;
40         cout << dis[a][b] << "\n";
41     }
42     return 0;
43 }
```

High Score (<https://cses.fi/problemset/task/1673>)

SPFA

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int v, w;
7 };
8 array<vector<E>, 2504> G, B;
9 array<bool, 2504> gis, bis;
10 array<int, 2504> dis, cnt;
11 void gfs(int u){
12     gis[u] = 1;
13     for(auto [v, w] : G[u]){
14         if(!gis[v]) gfs(v);
15     }
16 }
17 void bfs(int u){
18     bis[u] = 1;
19     for(auto [v, w] : B[u]){
20         if(!bis[v]) bfs(v);
21     }
22 }
23 int run(int s, int t){
24     int u;
25     queue<int> Q;
26     Q.push(s);
27     while(!Q.empty()){
28         u = Q.front();
29         Q.pop();
30         for(auto [v, w] : G[u]){
31             if(cnt[v] > t) continue;
32             if(dis[u] + w > dis[v]){
33                 Q.push(v);
34                 dis[v] = dis[u] + w;
35                 cnt[v]++;
36                 if(cnt[v] > t && gis[v] && bis[v]) return -1;
37             }
38         }
39     }
40     return dis[t];
41 }
42 signed main(){
43     int n, m, a, b, x;
44     cin >> n >> m;
45     for(int i = 2; i <= n; i++) dis[i] = -1e18;
46     while(m--){
47         cin >> a >> b >> x;
48         G[a].pb({b, x});
49         B[b].pb({a, x});
50     }
51     gfs(1);
52     bfs(n);
53     cout << run(1, n);
54     return 0;
55 }
```

Flight Discount (<https://cses.fi/problemset/task/1195>)

Dijkstra

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int v, w, c;
7 };
8 struct cmp{
9     bool operator()(E a, E b){
10         return a.w > b.w;
11     }
12 };
13 array<array<bool, 2>, 100004> vis;
14 array<vector<E>, 100004> G;
15 int run(int s, int t){
16     E e;
17     priority_queue<E, vector<E>, cmp> Q;
18     Q.push({s, 0, 0});
19     while(!Q.empty()){
20         e = Q.top();
21         Q.pop();
22         if(vis[e.v][e.c]) continue;
23         if(e.v == t) return e.w;
24         vis[e.v][e.c] = 1;
25         for(auto [v, w, c] : G[e.v]){
26             if(!(e.c & c)) Q.push({v, e.w + w, c | e.c});
27         }
28     }
29 }
30 signed main(){
31     int n, m, a, b, c;
32     cin >> n >> m;
33     while(m--){
34         cin >> a >> b >> c;
35         G[a].pb({b, c, 0});
36         G[a].pb({b, c / 2, 1});
37     }
38     cout << run(1, n);
39     return 0;
40 }
```

Cycle Finding (<https://cses.fi/problemset/task/1197>)

Bellman Ford

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int u, v, w;
7 };
8 vector<E> G;
9 array<int, 2504> dis, pre;
10 int run(int n){
11     int c;
12     for(int i = 0; i <= n; i++){
13         c = 0;
14         for(auto [u, v, w] : G){
15             if(dis[v] > dis[u] + w){
16                 dis[v] = dis[u] + w;
17                 pre[v] = u;
18                 c = v;
19             }
20         }
21     }
22     return c;
23 }
24 void print(int v, int u){
25     if(v == u){
26         cout << u << " ";
27         return;
28     }
29     print(pre[v], u);
30     cout << v << " ";
31 }
32 signed main(){
33     int n, m, a, b, c, u;
34     cin >> n >> m;
35     for(int i = 1; i <= n; i++){
36         dis[i] = 1e18;
37     }
38     while(m--){
39         cin >> a >> b >> c;
40         G.pb({a, b, c});
41     }
42     if(u = run(n)){
43         while(n--) u = pre[u];
44         cout << "YES\n";
45         print(pre[u], u);
46         cout << u;
47     }else cout << "NO";
48     return 0;
49 }
```

Flight Routes (<https://cses.fi/problemset/task/1196>)

Dijkstra

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int v, w;
7 };
8 struct cmp{
9     bool operator()(E a, E b){
10         return a.w > b.w;
11     }
12 };
13 array<vector<E>, 100004> G;
14 array<int, 100004> vis;
15 void run(int s, int t, int k){
16     E e;
17     priority_queue<E, vector<E>, cmp> Q;
18     Q.push({s, 0});
19     while(!Q.empty()){
20         e = Q.top();
21         Q.pop();
22         if(vis[e.v] >= k) continue;
23         if(e.v == t) cout << e.w << " ";
24         vis[e.v]++;
25         for(auto [v, w] : G[e.v]){
26             Q.push({v, w + e.w});
27         }
28     }
29 }
30 signed main(){
31     int n, m, k, a, b, c;
32     cin >> n >> m >> k;
33     while(m--){
34         cin >> a >> b >> c;
35         G[a].pb({b, c});
36     }
37     run(1, n, k);
38     return 0;
39 }
```

Round Trip II (<https://cses.fi/problemset/task/1678>)

DFS

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 100004> vis;
5 array<vector<int>, 100004> G;
6 stack<int> S;
7 vector<int> ans;
8 bool dfs(int u){
9     if(vis[u]) return 0;
10    vis[u] = 1;
11    S.push(u);
12    for(int v : G[u]){
13        if(vis[v] == 1){
14            while(S.top() != v){
15                ans.pb(v);
16                while(S.top() != v){
17                    ans.pb(S.top());
18                    S.pop();
19                }
20                ans.pb(v);
21            }
22            return 1;
23        }
24        if(dfs(v)) return 1;
25    }
26    S.pop();
27    vis[u] = 2;
28    return 0;
29 }
30 signed main(){
31     int n, m, a, b;
32     cin >> n >> m;
33     while(m--){
34         cin >> a >> b;
35         G[a].pb(b);
36     }
37     for(int i = 1; i <= n; i++){
38         if(dfs(i)){
39             cout << ans.size() << "\n";
40             reverse(ans.begin(), ans.end());
41             for(int a : ans) cout << a << " ";
42             return 0;
43         }
44     }
45     cout << "IMPOSSIBLE";
46     return 0;
47 }
```

Course Schedule (<https://cses.fi/problemset/task/1679>).

Topological Sort

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 100004> in;
5 array<vector<int>, 100004> G;
6 vector<int> ord;
7 void topu(int n){
8     int u;
9     queue<int> Q;
10    for(int i = 1; i <= n; i++){
11        if(!in[i]) Q.push(i);
12    }
13    while(!Q.empty()){
14        u = Q.front();
15        Q.pop();
16        ord.pb(u);
17        for(int v : G[u]){
18            in[v]--;
19            if(!in[v]) Q.push(v);
20        }
21    }
22 }
23 signed main(){
24     int n, m, a, b;
25     cin >> n >> m;
26     while(m--){
27         cin >> a >> b;
28         G[a].pb(b);
29         in[b]++;
30     }
31     topu(n);
32     if(ord.size() < n) cout << "IMPOSSIBLE";
33     else{
34         for(int o : ord) cout << o << " ";
35     }
36 }
37 }
```

[Longest Flight Route](https://cses.fi/problemset/task/1680) (<https://cses.fi/problemset/task/1680>)

Topological Sort DP

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<int, 100004> in, nxt, dp;
5 array<vector<int>, 100004> G;
6 stack<int> ord;
7 void topu(int n){
8     int u;
9     queue<int> Q;
10    dp[n] = 1;
11    for(int i = 1; i <= n; i++){
12        in[0]++;
13        if(!in[i]) Q.push(i);
14    }
15    while(!Q.empty()){
16        u = Q.front();
17        Q.pop();
18        ord.push(u);
19        for(int v : G[u]){
20            in[v]--;
21            if(!in[v]) Q.push(v);
22        }
23    }
24    while(!ord.empty()){
25        u = ord.top();
26        ord.pop();
27        for(int v : G[u]){
28            if(!dp[v]) continue;
29            if(dp[u] < dp[v] + 1){
30                dp[u] = dp[v] + 1;
31                nxt[u] = v;
32            }
33        }
34    }
35 }
36 void print(int u){
37     if(!u) return;
38     if(u == 1){
39         if(dp[1]) cout << dp[1] << "\n";
40         else{
41             cout << "IMPOSSIBLE";
42             return;
43         }
44     }
45     cout << u << " ";
46     print(nxt[u]);
47 }
48 signed main(){
49     int n, m, a, b;
50     cin >> n >> m;
51     while(m--){
52         cin >> a >> b;
53         in[b]++;
54         G[a].pb(b);
55     }
56     topu(n);
57     print(1);
58     return 0;
59 }
```

[Game Routes](https://cses.fi/problemset/task/1681) (<https://cses.fi/problemset/task/1681>)

Topological Sort DP

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 const int mod = 1e9 + 7;
5 array<int, 100004> in, dp;
6 array<vector<int>, 100004> G;
7 stack<int> ord;
8 int topu(int n){
9     int u;
10    queue<int> Q;
11    dp[n] = 1;
12    for(int i = 1; i <= n; i++){
13        if(!in[i]) Q.push(i);
14    }
15    while(!Q.empty()){
16        u = Q.front();
17        Q.pop();
18        ord.push(u);
19        for(int v : G[u]){
20            in[v]--;
21            if(!in[v]) Q.push(v);
22        }
23    }
24    while(!ord.empty()){
25        u = ord.top();
26        ord.pop();
27        for(int v : G[u]){
28            dp[u] += dp[v];
29            dp[u] %= mod;
30        }
31    }
32    return dp[1];
33 }
34 signed main(){
35     int n, m, a, b;
36     cin >> n >> m;
37     while(m--){
38         cin >> a >> b;
39         in[b]++;
40         G[a].pb(b);
41     }
42     cout << topu(n);
43     return 0;
44 }
```

[Investigation](https://cses.fi/problemset/task/1202) (<https://cses.fi/problemset/task/1202>)

Dijkstra DP

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 const int mod = 1e9 + 7;
6 struct E{
7     int v, w;
8 };
9 struct cmp{
10     bool operator()(E a, E b){
11         return a.w > b.w;
12     }
13 };
14 array<int, 100004> dis, dp, minfly, maxfly;
15 array<vector<E>, 100004> G, B;
16 vector<int> ord;
17 void run(int s){
18     E e;
19     dp[s] = 1;
20     minfly[s] = 1;
21     maxfly[s] = 1;
22     priority_queue<E, vector<E>, cmp> Q;
23     Q.push({s, 1});
24     dis[s] = 0;
25     while(!Q.empty()){
26         e = Q.top();
27         Q.pop();
28         if(dis[e.v]) continue;
29         ord.pb(e.v);
30         dis[e.v] = e.w;
31         for(auto [v, w] : B[e.v]){
32             Q.push({v, w + e.w});
33         }
34     }
35     for(int u : ord){
36         if(u != s) minfly[u] = 1e18;
37         for(auto [v, w] : G[u]){
38             if(dis[v] + w == dis[u]){
39                 dp[u] += dp[v];
40                 dp[u] %= mod;
41                 if(minfly[v]) minfly[u] = min(minfly[u], minfly[v] + 1);
42                 if(maxfly[v]) maxfly[u] = max(maxfly[u], maxfly[v] + 1);
43             }
44         }
45     }
46     cout << dis[1] - 1 << " " << dp[1] << " " << minfly[1] - 1 << " " << maxfly[1];
47 }
48 signed main(){
49     int n, m, a, b, c;
50     cin >> n >> m;
51     while(m--){
52         cin >> a >> b >> c;
53         G[a].pb({b, c});
54         B[b].pb({a, c});
55     }
56     run(n);
57     return 0;
58 }
```

[Planets Queries I](https://cses.fi/problemset/task/1750) (<https://cses.fi/problemset/task/1750>)

Doubling

```

1 #include <bits/stdc++.h>
2 using namespace std;
3 array<array<int, 30>, 200004> T;
4 void dabo(int n){
5     for(int j = 1; j <= 29; j++){
6         for(int i = 1; i <= n; i++){
7             T[i][j] = T[T[i][j - 1]][j - 1];
8         }
9     }
10 }
11 int find(int x, int k){
12     for(int i = 29; i >= 0; i--){
13         if((1 << i) & k){
14             x = T[x][i];
15             k -= (1 << i);
16         }
17     }
18     return x;
19 }
20 signed main(){
21     cin.tie(0), cout.tie(0), ios::sync_with_stdio(0);
22     int n, q, x, k;
23     cin >> n >> q;
24     for(int i = 1; i <= n; i++){
25         cin >> T[i][0];
26     }
27     dabo(n);
28     while(q--){
29         cin >> x >> k;
30         cout << find(x, k) << "\n";
31     }
32     return 0;
33 }
```

[Planets Queries II](https://cses.fi/problemset/task/1160) (<https://cses.fi/problemset/task/1160>)

DFS Doubling Topological Sort


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 int cnt = 0;
5 array<int, 200004> in, out, cyc, com, cycnt, T;
6 array<vector<int>, 200004> B;
7 array<array<int, 20>, 200004> G;
8 stack<int> ord;
9 int cfs(int u, int c, int x = 0){
10     if(com[u]) return x;
11     com[u] = c;
12     cyc[u] = ++x;
13     return cycnt[u] = cfs(T[u], c, x);
14 }
15 void dfs(int u, int c){
16     if(cyc[u]) in[u] = 0;
17     else in[u] = ++cnt;
18     com[u] = c;
19     for(int v : B[u]){
20         if(!in[v] && !cyc[v]) dfs(v, c);
21     }
22     if(cyc[u]) out[u] = 1e9;
23     else out[u] = ++cnt;
24 }
25 void topu(int n){
26     int u;
27     queue<int> Q;
28     for(int i = 1; i <= n; i++){
29         cyc[i] = 1;
30         if(!in[i]) Q.push(i);
31     }
32     while(!Q.empty()){
33         u = Q.front();
34         Q.pop();
35         ord.push(u);
36         cyc[u] = 0;
37         in[T[u]]--;
38         if(!in[T[u]]) Q.push(T[u]);
39     }
40     for(int i = 1; i <= n; i++){
41         if(cyc[i]){
42             cfs(i, i);
43             dfs(i, com[i] ? com[i] : i);
44         }
45     }
46     while(!ord.empty()){
47         u = ord.top();
48         ord.pop();
49         if(!com[u]) dfs(u, u);
50     }
51 }
52 void dabo(int n){
53     for(int j = 1; j < 20; j++){
54         for(int i = 1; i <= n; i++){
55             G[i][j] = G[G[i][j - 1]][j - 1];
56         }
57     }
58 }
59 int query(int a, int b){
60     if(a == b) return 0;
61     if(com[a] != com[b]) return -1;
62     int ans = 0;
63     for(int i = 19; i >= 0; i--){
64         if(in[G[a][i]] > in[b] && out[G[a][i]] < out[b]){
65             a = G[a][i];
```

```
66         ans += 1 << i;
67     }
68 }
69 a = G[a][0];
70 ans++;
71 if(a == b) return ans;
72 if(cyc[a] && cyc[b]){
73     ans += (cyc[b] - cyc[a] + cycnt[a]) % cycnt[b];
74     return ans;
75 }
76 return -1;
77 }
78 signed main(){
79     int n, q, a, b;
80     cin >> n >> q;
81     for(int i = 1; i <= n; i++){
82         cin >> T[i];
83         in[T[i]]++;
84         if(i == T[i]) in[T[i]]--;
85         G[i][0] = T[i];
86         B[T[i]].pb(i);
87     }
88     topu(n);
89     dabo(n);
90     while(q--){
91         cin >> a >> b;
92         cout << query(a, b) << "\n";
93     }
94     return 0;
95 }
```

Planets Cycles (<https://cses.fi/problemset/task/1751>).

DFS Doubling Topological Sort


```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 int cnt = 0;
5 array<int, 200004> in, out, cyc, com, cycnt, T;
6 array<vector<int>, 200004> B;
7 array<array<int, 20>, 200004> G;
8 stack<int> ord;
9 int cfs(int u, int c, int x = 0){
10     if(com[u]) return x;
11     com[u] = c;
12     cyc[u] = ++x;
13     return cycnt[u] = cfs(T[u], c, x);
14 }
15 void dfs(int u, int c){
16     if(cyc[u]) in[u] = 0;
17     else in[u] = ++cnt;
18     com[u] = c;
19     for(int v : B[u]){
20         if(!in[v] && !cyc[v]) dfs(v, c);
21     }
22     if(cyc[u]) out[u] = 1e9;
23     else out[u] = ++cnt;
24 }
25 void topu(int n){
26     int u;
27     queue<int> Q;
28     for(int i = 1; i <= n; i++){
29         cyc[i] = 1;
30         if(!in[i]) Q.push(i);
31     }
32     while(!Q.empty()){
33         u = Q.front();
34         Q.pop();
35         ord.push(u);
36         cyc[u] = 0;
37         in[T[u]]--;
38         if(!in[T[u]]) Q.push(T[u]);
39     }
40     for(int i = 1; i <= n; i++){
41         if(cyc[i]){
42             cfs(i, i);
43             dfs(i, com[i] ? com[i] : i);
44         }
45     }
46     while(!ord.empty()){
47         u = ord.top();
48         ord.pop();
49         if(!com[u]) dfs(u, u);
50     }
51 }
52 void dabo(int n){
53     in[0] = 0, out[0] = 1e9;
54     for(int j = 1; j < 20; j++){
55         for(int i = 1; i <= n; i++){
56             G[i][j] = G[G[i][j - 1]][j - 1];
57         }
58     }
59 }
60 int query(int x){
61     int ans = 0;
62     if(T[x] == x) return 1;
63     if(cyc[x]) return cycnt[x];
64     for(int i = 19; i >= 0; i--){
65         if(in[G[x][i]] > in[0] && out[G[x][i]] < out[0]){

```

```
66         x = G[x][i];
67         ans += 1 << i;
68     }
69 }
70 x = G[x][0];
71 ans++;
72 if(cyc[x]) return ans + cycnt[x];
73 return -1;
74 }
75 signed main(){
76     int n, q, a, b;
77     cin >> n;
78     for(int i = 1; i <= n; i++){
79         cin >> T[i];
80         in[T[i]]++;
81         G[i][0] = T[i];
82         B[T[i]].pb(i);
83     }
84     topu(n);
85     dabo(n);
86     for(int i = 1; i <= n; i++){
87         cout << query(i) << " ";
88     }
89     return 0;
90 }
```

Road Reparation (<https://cses.fi/problemset/task/1675>)

Kruskal

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct E{
6     int u, v, w;
7 };
8 array<int, 100004> DSU;
9 vector<E> G;
10 bool cmp(E a, E b){
11     return a.w < b.w;
12 }
13 int find(int x){
14     if(DSU[x] == x) return x;
15     return DSU[x] = find(DSU[x]);
16 }
17 void onion(int a, int b){
18     int A = find(a), B = find(b);
19     DSU[B] = A;
20 }
21 void MST(int n){
22     int ans = 0, cnt = n;
23     for(auto [v, u, w] : G){
24         if(find(v) == find(u)) continue;
25         cnt--;
26         onion(u, v);
27         ans += w;
28     }
29     if(cnt == 1) cout << ans;
30     else cout << "IMPOSSIBLE";
31 }
32 signed main(){
33     int n, m, a, b, c;
34     cin >> n >> m;
35     for(int i = 1; i <= n; i++){
36         DSU[i] = i;
37     }
38     while(m--){
39         cin >> a >> b >> c;
40         G.pb({a, b, c});
41     }
42     sort(G.begin(), G.end(), cmp);
43     MST(n);
44     return 0;
45 }
```

Road Construction (<https://cses.fi/problemset/task/1676>)

DSU

```
1 #include <bits/stdc++.h>
2 using namespace std;
3 array<int, 100004> DSU, S;
4 int find(int x){
5     if(DSU[x] == x) return x;
6     return DSU[x] = find(DSU[x]);
7 }
8 int onion(int a, int b){
9     int A = find(a), B = find(b);
10    if(S[A] < S[B]){
11        DSU[A] = B;
12        S[B] += S[A];
13        return S[B];
14    }else{
15        DSU[B] = A;
16        S[A] += S[B];
17        return S[A];
18    }
19 }
20 signed main(){
21     int n, m, a, b, cnt, sz = 1;
22     cin >> n >> m;
23     cnt = n;
24     for(int i = 1; i <= n; i++){
25         DSU[i] = i;
26         S[i] = 1;
27     }
28     while(m--){
29         cin >> a >> b;
30         if(find(a) != find(b)){
31             cnt--;
32             sz = max(sz, onion(a, b));
33         }
34         cout << cnt << " " << sz << "\n";
35     }
36     return 0;
37 }
```

Flight Routes Check (<https://cses.fi/problemset/task/1682>)

SCC

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 int cnt = 0;
5 array<bool, 100004> vis;
6 array<vector<int>, 100004> G, B, S;
7 array<int, 100004> scc, deg;
8 stack<int> out;
9 void bfs(int u){
10     if(vis[u]) return;
11     vis[u] = 1;
12     for(int v : B[u]){
13         bfs(v);
14     }
15     out.push(u);
16 }
17 void dfs(int u, int s){
18     if(scc[u]) return;
19     scc[u] = s;
20     for(int v : G[u]){
21         dfs(v, s);
22     }
23 }
24 signed main(){
25     int n, m, a, b, o;
26     cin >> n >> m;
27     while(m--){
28         cin >> a >> b;
29         G[a].pb(b);
30         B[b].pb(a);
31     }
32     for(int i = 1; i <= n; i++){
33         bfs(i);
34     }
35     while(!out.empty()){
36         o = out.top();
37         out.pop();
38         if(!scc[o]) dfs(o, ++cnt);
39     }
40     for(int i = 1; i <= n; i++){
41         S[scc[i]].pb(i);
42         for(int v : G[i]){
43             if(scc[v] == scc[i]) continue;
44             deg[scc[i]]++;
45         }
46     }
47     for(int i = 1; i <= cnt; i++){
48         if(deg[i]) b = S[i][0];
49         else a = S[i][0];
50     }
51     if(cnt == 1) cout << "YES";
52     else{
53         cout << "NO\n" << a << " " << b;
54     }
55     return 0;
56 }
```

Planets and Kingdoms (<https://cses.fi/problemset/task/1683>)

SCC

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 int k = 0;
5 array<vector<int>, 100004> G, B;
6 array<bool, 100004> vis;
7 array<int, 100004> scc;
8 stack<int> out;
9 void bfs(int u){
10     if(vis[u]) return;
11     vis[u] = 1;
12     for(int v : B[u]){
13         bfs(v);
14     }
15     out.push(u);
16 }
17 void dfs(int u){
18     if(scc[u]) return;
19     scc[u] = k;
20     for(int v : G[u]){
21         dfs(v);
22     }
23 }
24 signed main(){
25     int n, m, a, b, o;
26     cin >> n >> m;
27     while(m--){
28         cin >> a >> b;
29         G[a].pb(b);
30         B[b].pb(a);
31     }
32     for(int i = 1; i <= n; i++){
33         bfs(i);
34     }
35     while(!out.empty()){
36         o = out.top();
37         out.pop();
38         if(!scc[o]) ++k, dfs(o);
39     }
40     cout << k << "\n";
41     for(int i = 1; i <= n; i++){
42         cout << scc[i] << " ";
43     }
44     return 0;
45 }
```

Giant Pizza (<https://cses.fi/problemset/task/1684>)

2-SAT Topological Sort


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 int k = 0;
5 array<vector<int>, 200004> G, B, C, S;
6 array<bool, 200004> vis;
7 array<int, 200004> scc, ans, in;
8 stack<int> out, ord;
9 void bfs(int u){
10     if(vis[u]) return;
11     vis[u] = 1;
12     for(int v : B[u]) bfs(v);
13     out.push(u);
14 }
15 void dfs(int u){
16     if(scc[u]) return;
17     scc[u] = k;
18     for(int v : G[u]) dfs(v);
19 }
20 void topu(int n){
21     int u;
22     bool ok;
23     queue<int> Q;
24     for(int i = 1; i <= 2 * n + 1; i++){
25         if(!in[i]) Q.push(i);
26         ans[i] = -1;
27     }
28     while(!Q.empty()){
29         u = Q.front();
30         Q.pop();
31         ord.push(u);
32         for(int v : S[u]){
33             in[v]--;
34             if(!in[v]) Q.push(v);
35         }
36     }
37     while(!ord.empty()){
38         u = ord.top();
39         ord.pop();
40         ok = 1;
41         for(int v : C[u]){
42             if(ans[v / 2] >= 0) ok = 0;
43         }
44         if(!ok) continue;
45         for(int v : C[u]){
46             ans[v / 2] = v & 1;
47         }
48     }
49 }
50 signed main(){
51     int n, m, a, b, o;
52     char wa, wb;
53     bool ok = 1;
54     cin >> m >> n;
55     while(m--){
56         cin >> wa >> a >> wb >> b;
57         a = 2 * a;
58         b = 2 * b;
59         if(wa == '+') a++;
60         if(wb == '+') b++;
61         G[a ^ 1].pb(b);
62         B[b].pb(a ^ 1);
63         G[b ^ 1].pb(a);
64         B[a].pb(b ^ 1);
65     }
}
```

```
66     for(int i = 2; i <= 2 * n + 1; i++){
67         bfs(i);
68     }
69     while(!out.empty()){
70         o = out.top();
71         out.pop();
72         if(!scc[o]) k++, dfs(o);
73     }
74     for(int i = 1; i <= n; i++){
75         //cout << scc[2 * i] << " " << scc[2 * i + 1] << "\n";
76         if(scc[2 * i] == scc[2 * i + 1]) ok = 0;
77     }
78     if(!ok){
79         cout << "IMPOSSIBLE";
80         return 0;
81     }
82     for(int i = 2; i <= 2 * n + 1; i++){
83         C[scc[i]].pb(i);
84         for(int v : G[i]){
85             if(scc[v] == scc[i]) continue;
86             in[scc[v]]++;
87             S[scc[i]].pb(scc[v]);
88         }
89     }
90     topu(n);
91     for(int i = 1; i <= n; i++){
92         cout << (ans[i] ? "+" : "-");
93     }
94     return 0;
95 }
```

Coin Collector (<https://cses.fi/problemset/task/1686>).

SCC Topological Sort DP


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 int k = 0;
6 array<int, 100004> K, scc, dp, val, in;
7 array<bool, 100004> vis;
8 array<vector<int>, 100004> G, B, S;
9 stack<int> out, ord;
10 void bfs(int u){
11     if(vis[u]) return;
12     vis[u] = 1;
13     for(int v : B[u]) bfs(v);
14     out.push(u);
15 }
16 void dfs(int u){
17     if(scc[u]) return;
18     scc[u] = k;
19     for(int v : G[u]) dfs(v);
20 }
21 int topu(int n){
22     int u, ans = 0;
23     queue<int> Q;
24     for(int i = 1; i <= n; i++){
25         if(!in[i]) Q.push(i);
26     }
27     while(!Q.empty()){
28         u = Q.front();
29         Q.pop();
30         ord.push(u);
31         for(int v : S[u]){
32             in[v]--;
33             if(!in[v]) Q.push(v);
34         }
35     }
36     while(!ord.empty()){
37         u = ord.top();
38         ord.pop();
39         dp[u] = val[u];
40         for(int v : S[u]){
41             dp[u] = max(dp[u], dp[v] + val[u]);
42         }
43         ans = max(ans, dp[u]);
44     }
45     return ans;
46 }
47 signed main(){
48     int n, m, a, b, o;
49     cin >> n >> m;
50     for(int i = 1; i <= n; i++){
51         cin >> K[i];
52     }
53     while(m--){
54         cin >> a >> b;
55         G[a].pb(b);
56         B[b].pb(a);
57     }
58     for(int i = 1; i <= n; i++){
59         bfs(i);
60     }
61     while(!out.empty()){
62         o = out.top();
63         out.pop();
64         if(!scc[o]) k++, dfs(o);
65     }
}
```

```

66     for(int i = 1; i <= n; i++){
67         val[scc[i]] += K[i];
68         for(int v : G[i]){
69             if(scc[v] == scc[i]) continue;
70             in[scc[v]]++;
71             S[scc[i]].pb(scc[v]);
72         }
73     }
74     cout << topu(k);
75     return 0;
76 }
```

Mail Delivery (<https://cses.fi/problemset/task/1691>)

Euler Tour

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<set<int>, 100004> G;
5 vector<int> ans;
6 void ola(int u){
7     int v;
8     while(G[u].size()){
9         v = *G[u].begin();
10        G[u].erase(v);
11        G[v].erase(u);
12        ola(v);
13    }
14    ans.pb(u);
15 }
16 signed main(){
17     int n, m, a, b;
18     cin >> n >> m;
19     for(int i = 0; i < m; i++){
20         cin >> a >> b;
21         G[a].insert(b);
22         G[b].insert(a);
23     }
24     for(int i = 1; i <= n; i++){
25         if(G[i].size() & 1){
26             cout << "IMPOSSIBLE";
27             return 0;
28         }
29     }
30     ola(1);
31     if(ans.size() < m + 1){
32         cout << "IMPOSSIBLE";
33         return 0;
34     }
35     reverse(ans.begin(), ans.end());
36     for(int a : ans) cout << a << " ";
37     return 0;
38 }
```

De Bruijn Sequence (<https://cses.fi/problemset/task/1692>)

DFS

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define ppb pop_back
4 using namespace std;
5 int n;
6 array<array<int, 2>, 1 << 15> G;
7 array<bool, 1 << 15> vis;
8 vector<int> ans;
9 int dfs(int u, int cnt){
10     if(vis[u]) return u;
11     int v = 0, tmp;
12     vis[u] = 1;
13     if(cnt == 1 << n){
14         for(int i = 0; i < n; i++) cout << "0";
15         for(int a : ans) cout << a;
16         exit(0);
17     }
18     for(int i = 0; i < 2; i++){
19         if(G[u][i] == -1) continue;
20         ans.pb(i);
21         tmp = dfs(G[u][i], cnt + 1);
22         if(tmp && tmp != u) v = tmp, G[u][i] = -1;
23         ans.pp();
24     }
25     vis[u] = 0;
26     return v;
27 }
28 signed main(){
29     cin >> n;
30     for(int i = 0; i < 1 << n; i++){
31         G[i][0] = (i << 1) & ((1 << n) - 1);
32         G[i][1] = ((i << 1) | 1) & ((1 << n) - 1);
33     }
34     dfs(0, 1);
35     return 0;
36 }
```

Teleporters Path (<https://cses.fi/problemset/task/1693>)

Euler Tour

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<set<int>, 100004> G;
5 array<int, 100004> in;
6 stack<int> ans;
7 void ola(int u){
8     int v;
9     while(G[u].size()){
10         v = *G[u].begin();
11         G[u].erase(v);
12         ola(v);
13     }
14     ans.push(u);
15 }
16 signed main(){
17     int n, m, a, b;
18     cin >> n >> m;
19     for(int i = 0; i < m; i++){
20         cin >> a >> b;
21         G[a].insert(b);
22         in[b]++;
23     }
24     in[a]++;
25     in[n]--;
26     for(int i = 1; i <= n; i++){
27         if(in[i] != G[i].size()){
28             cout << "IMPOSSIBLE";
29             return 0;
30         }
31     }
32     ola(1);
33     if(ans.size() == m + 1){
34         while(!ans.empty()){
35             cout << ans.top() << " ";
36             ans.pop();
37         }
38     }else cout << "IMPOSSIBLE";
39     return 0;
40 }
```

Hamiltonian Flights (<https://cses.fi/problemset/task/1690>)

Hamiltonian Cycle 狀壓DP

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define pii pair<int, int>
4 #define ff first
5 #define ss second
6 using namespace std;
7 const int mod = 1e9 + 7;
8 array<vector<int>, 24> G;
9 array<array<int, 20>, 1 << 20> dp;
10 array<array<bool, 20>, 1 << 20> vis;
11 int run(int n){
12     int s, u;
13     queue<pii> Q;
14     dp[1][0] = 1;
15     Q.push({1, 0});
16     vis[1][0] = 1;
17     while(!Q.empty()){
18         s = Q.front().ff;
19         u = Q.front().ss;
20         Q.pop();
21         if(u == n - 1) continue;
22         for(int v : G[u]){
23             if((1 << v) & s) continue;
24             dp[s | (1 << v)][v] += dp[s][u];
25             dp[s | (1 << v)][v] %= mod;
26             if(vis[s | (1 << v)][v]) continue;
27             Q.push({s | (1 << v), v});
28             vis[s | (1 << v)][v] = 1;
29         }
30     }
31     return dp[(1 << n) - 1][n - 1];
32 }
33 signed main(){
34     int n, m, a, b;
35     cin >> n >> m;
36     while(m--){
37         cin >> a >> b;
38         a--, b--;
39         G[a].pb(b);
40     }
41     cout << run(n);
42     /*for(int i = 1; i < 1 << n; i++){
43         for(int j = 0; j < n; j++){
44             cout << dp[i][j] << " ";
45         }
46         cout << "\n";
47     }*/
48     return 0;
49 }
```

Knight's Tour (<https://cses.fi/problemset/task/1689>)

暴力 DFS

```

1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 struct V{
5     int d, x, y;
6 };
7 array<int, 8> dx = {-2, -1, 1, 2, 2, 1, -1, -2}, dy = {1, 2, 2, 1, -1, -2, -2,
8 array<array<int, 12>, 12> C;
9 int deg(int x, int y){
10     int d = 0, nx, ny;
11     for(int i = 0; i < 8; i++){
12         nx = x + dx[i], ny = y + dy[i];
13         if(nx < 1 || ny < 1 || nx > 8 || ny > 8 || C[nx][ny]) continue;
14         d++;
15     }
16     return d;
17 }
18 void dfs(int x, int y, int s){
19     C[x][y] = s;
20     if(s == 64){
21         for(int i = 1; i <= 8; i++){
22             for(int j = 1; j <= 8; j++){
23                 cout << C[i][j] << " ";
24             }
25             cout << "\n";
26         }
27         exit(0);
28     }
29     int nx, ny;
30     vector<V> nxt;
31     for(int i = 0; i < 8; i++){
32         nx = x + dx[i], ny = y + dy[i];
33         if(nx < 1 || ny < 1 || nx > 8 || ny > 8 || C[nx][ny]) continue;
34         nxt.pb({deg(nx, ny), nx, ny});
35     }
36     sort(nxt.begin(), nxt.end(), [] (V a, V b){return a.d < b.d;});
37     for(V v : nxt){
38         dfs(v.x, v.y, s + 1);
39     }
40     C[x][y] = 0;
41 }
42 signed main(){
43     int x, y;
44     cin >> x >> y;
45     dfs(y, x, 1);
46     return 0;
47 }
```

Download Speed (<https://cses.fi/problemset/task/1694>)

Dinic


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define int long long
4 using namespace std;
5 struct pipe{
6     int u, v, f;
7 };
8 const int inf = 4e12;
9 int t;
10 array<int, 504> lvl, P;
11 array<vector<int>, 504> G;
12 vector<pipe> E;
13 int bfs(int s){
14     int u, v;
15     queue<int> Q;
16     Q.push(s);
17     lvl[s] = 1;
18     while(!Q.empty()){
19         u = Q.front();
20         Q.pop();
21         for(int i : G[u]){
22             v = E[i].v;
23             if(lvl[v] || !E[i].f) continue;
24             lvl[v] = lvl[u] + 1;
25             Q.push(v);
26         }
27     }
28     return lvl[t];
29 }
30 int dfs(int u, int f){
31     if(u == t || !f) return f;
32     int wut, ans = 0;
33     for(int &i = P[u]; i < G[u].size(); i++){
34         pipe &e = E[G[u][i]], &b = E[G[u][i] ^ 1];
35         if(lvl[e.v] == lvl[u] + 1){
36             wut = dfs(e.v, min(e.f, f));
37             e.f -= wut;
38             b.f += wut;
39             f -= wut;
40             ans += wut;
41         }
42     }
43     return ans;
44 }
45 int dinic(int s){
46     int ans = 0, tmp;
47     while(1){
48         for(int &l : lvl) l = 0;
49         if(!bfs(s)) break;
50         while(1){
51             for(int &p : P) p = 0;
52             if(tmp = dfs(s, inf)) ans += tmp;
53             else break;
54         }
55     }
56     return ans;
57 }
58 signed main(){
59     int n, m, a, b, c, cnt = 0;
60     cin >> n >> m;
61     while(m--){
62         cin >> a >> b >> c;
63         G[a].pb(cnt++);
64         G[b].pb(cnt++);
65         E.pb({a, b, c});
66 }
```

```
66     E.pb({b, a, 0});  
67 }  
68 t = n;  
69 cout << dinic(1);  
70 return 0;  
71 }
```

Police Chase (<https://cses.fi/problemset/task/1695>)

Dinic


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 struct pipe{
5     int u, v, f;
6 };
7 const int inf = 1e3;
8 int t;
9 array<int, 504> lvl, P, vis;
10 array<vector<int>, 504> G;
11 vector<pipe> E;
12 int bfs(int s){
13     int u, v;
14     queue<int> Q;
15     Q.push(s);
16     lvl[s] = 1;
17     while(!Q.empty()){
18         u = Q.front();
19         Q.pop();
20         for(int i : G[u]){
21             v = E[i].v;
22             if(lvl[v] || !E[i].f) continue;
23             lvl[v] = lvl[u] + 1;
24             Q.push(v);
25         }
26     }
27     return lvl[t];
28 }
29 int dfs(int u, int f){
30     if(u == t || !f) return f;
31     int wut, ans = 0;
32     for(int &i = P[u]; i < G[u].size(); i++){
33         pipe &e = E[G[u][i]], &b = E[G[u][i]] ^ 1;
34         if(lvl[e.v] == lvl[u] + 1){
35             wut = dfs(e.v, min(e.f, f));
36             e.f -= wut;
37             b.f += wut;
38             f -= wut;
39             ans += wut;
40         }
41     }
42     return ans;
43 }
44 int dinic(int s){
45     int ans = 0, tmp;
46     while(1){
47         for(int &l : lvl) l = 0;
48         if(!bfs(s)) break;
49         while(1){
50             for(int &p : P) p = 0;
51             if(tmp = dfs(s, inf)) ans += tmp;
52             else break;
53         }
54     }
55     return ans;
56 }
57 void go(int u, int c){
58     if(vis[u]) return;
59     vis[u] = c;
60     for(int i : G[u]){
61         if(!E[i].f) continue;
62         go(E[i].v, c);
63     }
64 }
65 signed main(){
```

```
66     int n, m, a, b, cnt = 0;
67     cin >> n >> m;
68     while(m--){
69         cin >> a >> b;
70         G[a].pb(cnt++);
71         G[b].pb(cnt++);
72         E.pb({a, b, 1});
73         E.pb({b, a, 1});
74     }
75     t = n;
76     cout << dinic(1) << "\n";
77     go(1, 1);
78     go(t, 2);
79     for(pipe e : E){
80         if(e.f) continue;
81         if(vis[e.u] && vis[e.v] && vis[e.u] != vis[e.v]){
82             cout << e.u << " " << e.v << "\n";
83         }
84     }
85     return 0;
86 }
```

School Dance (<https://cses.fi/problemset/task/1696>)

二分圖匹配

```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 using namespace std;
4 array<bool, 504> vis;
5 array<int, 504> mate;
6 array<vector<int>, 504> G;
7 int rob(int u){
8     for(int v : G[u]){
9         if(vis[v]) continue;
10        vis[v] = 1;
11        if(!mate[v] || rob(mate[v])){
12            mate[v] = u;
13            return 1;
14        }
15    }
16    return 0;
17 }
18 signed main(){
19     int n, m, k, a, b, cnt = 0;
20     cin >> n >> m >> k;
21     while(k--){
22         cin >> a >> b;
23         G[a].pb(b);
24     }
25     for(int i = 1; i <= n; i++){
26         for(bool &v : vis) v = 0;
27         cnt += rob(i);
28     }
29     cout << cnt << "\n";
30     for(int i = 1; i <= m; i++){
31         if(mate[i]){
32             cout << mate[i] << " " << i << "\n";
33         }
34     }
35     return 0;
36 }
```

Distinct Routes (<https://cses.fi/problemset/task/1711>)

Dinic


```
1 #include <bits/stdc++.h>
2 #define pb push_back
3 #define ppb pop_back
4 using namespace std;
5 struct pipe{
6     int u, v, f;
7 };
8 const int inf = 1e3;
9 int t;
10 array<bool, 504> vis;
11 array<int, 504> lvl, P;
12 array<vector<int>, 504> G;
13 vector<int> ans;
14 vector<pipe> E;
15 int bfs(int s){
16     int u, v;
17     queue<int> Q;
18     Q.push(s);
19     lvl[s] = 1;
20     while(!Q.empty()){
21         u = Q.front();
22         Q.pop();
23         for(int i : G[u]){
24             v = E[i].v;
25             if(lvl[v] || !E[i].f) continue;
26             lvl[v] = lvl[u] + 1;
27             Q.push(v);
28         }
29     }
30     return lvl[t];
31 }
32 int dfs(int u, int f){
33     if(u == t || !f) return f;
34     int wut, ans = 0;
35     for(int &i = P[u]; i < G[u].size(); i++){
36         pipe &e = E[G[u][i]], &b = E[G[u][i] ^ 1];
37         if(lvl[e.v] == lvl[u] + 1){
38             wut = dfs(e.v, min(e.f, f));
39             e.f -= wut;
40             b.f += wut;
41             f -= wut;
42             ans += wut;
43         }
44     }
45     return ans;
46 }
47 int dinic(int s){
48     int ans = 0, tmp;
49     while(1){
50         for(int &l : lvl) l = 0;
51         if(!bfs(s)) break;
52         while(1){
53             for(int &p : P) p = 0;
54             if(tmp = dfs(s, inf)) ans += tmp;
55             else break;
56         }
57     }
58     return ans;
59 }
60 bool run(int u){
61     if(vis[u]) return 0;
62     ans.pb(u);
63     if(u == t){
64         cout << ans.size() << "\n";
65         for(int a : ans) cout << a << " ";
66 }
```

```
66         cout << "\n";
67         return 1;
68     }
69     vis[u] = 1;
70     for(int i : G[u]){
71         if(i & 1 || E[i].f) continue;
72         if(run(E[i].v)){
73             E[i].f++;
74             vis[u] = 0;
75             return 1;
76         }
77     }
78     ans.ppb();
79     vis[u] = 0;
80     return 0;
81 }
82 signed main(){
83     int n, m, a, b, cnt = 0;
84     cin >> n >> m;
85     while(m--){
86         cin >> a >> b;
87         G[a].pb(cnt++);
88         G[b].pb(cnt++);
89         E.pb({a, b, 1});
90         E.pb({b, a, 0});
91     }
92     t = n;
93     cout << (cnt = dinic(1)) << "\n";
94     while(cnt--){
95         ans.clear();
96         run(1);
97     }
98     return 0;
99 }
```