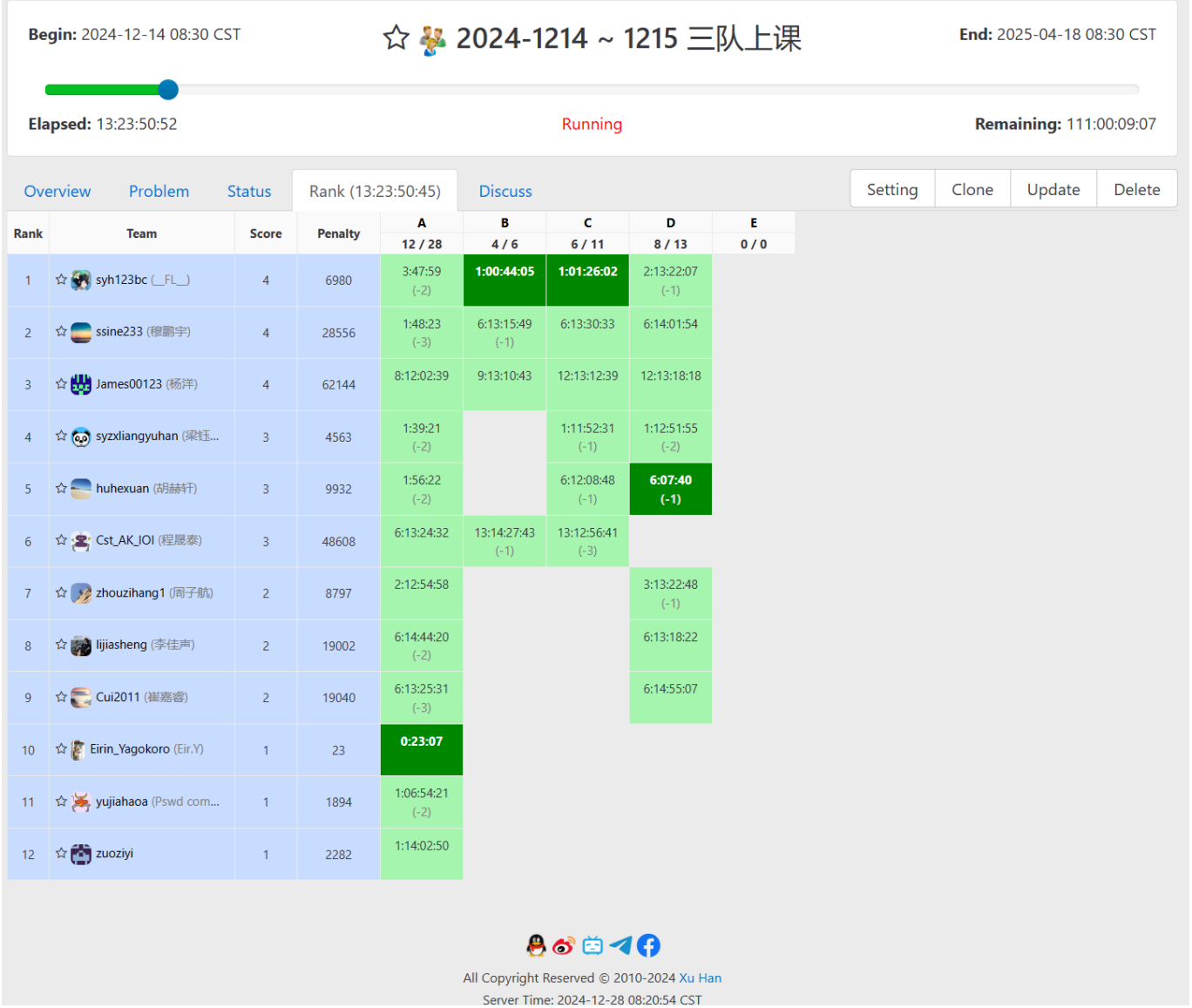
思考题讲解

人员

于珈浩、杨洋、左子毅 到课

上周作业检查



Begin: 2024-12-14 08:30 CST

☆ 2024-1214 ~ 1215 三队上课

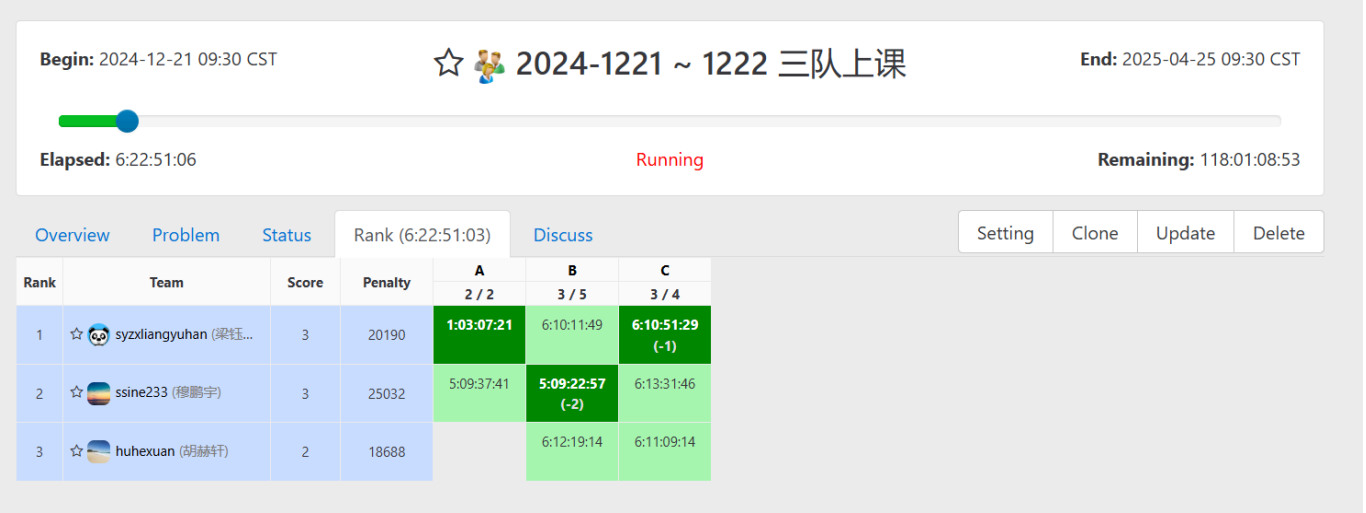
End: 2025-04-18 08:30 CST

Elapsed: 13:23:50:52 Running **Remaining:** 111:00:09:07

Overview Problem Status Rank (13:23:50:45) Discuss Setting Clone Update Delete

Rank	Team	Score	Penalty	A 12 / 28	B 4 / 6	C 6 / 11	D 8 / 13	E 0 / 0
1	☆ syh123bc (_FL_)	4	6980	3:47:59 (-2)	**1:00:44:05**	**1:01:26:02**	2:13:22:07 (-1)	
2	☆ ssine233 (穆鹏宇)	4	28556	1:48:23 (-3)	6:13:15:49 (-1)	6:13:30:33	6:14:01:54	
3	☆ James00123 (杨洋)	4	62144	8:12:02:39	9:13:10:43	12:13:12:39	12:13:18:18	
4	☆ syzxliangyuhan (梁钰...	3	4563	1:39:21 (-2)		1:11:52:31 (-1)	1:12:51:55 (-2)	
5	☆ huhexuan (胡赫轩)	3	9932	1:56:22 (-2)		6:12:08:48 (-1)	**6:07:40 (-1)**	
6	☆ Cst_AK_IOI (程晟泰)	3	48608	6:13:24:32	13:14:27:43 (-1)	13:12:56:41 (-3)		
7	☆ zhouzihang1 (周子航)	2	8797	2:12:54:58			3:13:22:48 (-1)	
8	☆ lijiasheng (李佳声)	2	19002	6:14:44:20 (-2)			6:13:18:22	
9	☆ Cui2011 (崔嘉睿)	2	19040	6:13:25:31 (-3)			6:14:55:07	
10	☆ Eirin_Yagokoro (Eir.Y)	1	23	**0:23:07**				
11	☆ yujiahaoa (Pswd com...	1	1894	1:06:54:21 (-2)				
12	☆ zuoziyi	1	2282	1:14:02:50				

Server Time: 2024-12-28 08:20:54 CST



作业

https://vjudge.net/contest/682535

课堂表现

今天课上讲的题目比较多，同学们课下一定要好好写一下每道题目, 彻底理解每一道题目。

课堂内容

CF222E Decoding Genome

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

using namespace std;

int get_int(char x) { return (islower(x) ? x-'a'+1 : x-'A'+27); }

typedef long long LL;
const int mod = 1e9 + 7;
const int maxn = 52 + 5;

struct Matrix {
  int c[maxn][maxn];
  int n, m;

  Matrix(): n(0), m(0) { memset(c, 0, sizeof(c)); }
  Matrix(int n_): n(n_), m(n_) { memset(c, 0, sizeof(c)); }
  Matrix(int n_, int m_): n(n_), m(m_) { memset(c, 0, sizeof(c)); }

  Matrix operator * (const Matrix& p) const {
    Matrix res(n, p.m);
    for (int i = 1; i <= n; ++i) {
      for (int j = 1; j <= p.m; ++j) {
        for (int k = 1; k <= m; ++k) {
          res.c[i][j] = (1LL*c[i][k]*p.c[k][j] + res.c[i][j]) % mod;
        }
```

```
      }
    }
    return res;
  }

  void build(int n_) {
    n = m = n_; memset(c, 0, sizeof(c));
    for (int i = 1; i <= n; ++i) c[i][i] = 1;
  }
};

Matrix qmod(Matrix a, LL k) {
  Matrix res; res.build(a.n);
  while (k) {
    if (k&1) res = res * a;
    a = a * a;
    k >>= 1;
  }
  return res;
}

int main()
{
  LL n; int m, k; cin >> n >> m >> k;

  Matrix mtx(m);
  for (int i = 1; i <= m; ++i) {
    for (int j = 1; j <= m; ++j) mtx.c[i][j] = 1;
  }

  while (k -- ) {
    string s; cin >> s;
    int a = get_int(s[0]), b = get_int(s[1]);
    mtx.c[a][b] = 0;
  }

  mtx = qmod(mtx, n-1);

  Matrix f(1, m);
  for (int i = 1; i <= m; ++i) f.c[1][i] = 1;
  f = f * mtx;

  int res = 0;
  for (int i = 1; i <= m; ++i) res = (res + f.c[1][i]) % mod;
  cout << res << endl;
  return 0;
}
```

CF1353E K-periodic Garland

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

using namespace std;

const int maxn = 1e6 + 5;
const int inf = 0x3f3f3f3f;
char s[maxn];
int f[maxn][2], p[maxn];

int get_sum(int l, int r) { return p[r] - p[l-1]; }

void solve() {
  int n, k; cin >> n >> k;
  for (int i = 0; i <= n+2; ++i) {
    f[i][0] = 0, f[i][1] = inf, p[i] = 0;
  }

  cin >> (s+1);
  for (int i = 1; i <= n; i++) p[i] = p[i-1] + (s[i]=='1');
  for (int i = 1; i <= n; ++i) {
    f[i][0] = p[i-1] + (s[i]!='1');
    if (i >= k+1) f[i][1] = min(f[i-k][0], f[i-k][1]) + get_sum(i-k+1, i-1) +
(s[i]!='1');
  }

  int res = get_sum(1, n);
  for (int i = 1; i <= n; ++i) {
    res = min(res, min(f[i][0], f[i][1]) + get_sum(i+1, n));
  }

//  cout << " ------------------ ";
  cout << res << endl;
}

int main()
{
  ios::sync_with_stdio(false);
  cin.tie(0), cout.tie(0);

  int T; cin >> T;
  while (T -- ) solve();
  return 0;
}
```

CF547B Mike and Feet

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

using namespace std;
```

```
const int maxn = 2e5 + 5;
int w[maxn], pre[maxn], suf[maxn];
int f[maxn];

void solve_pre(int n) {
  stack<int> s;
  for (int i = 1; i <= n; ++i) {
    while (!s.empty() && w[i]<=w[s.top()]) s.pop();
    pre[i] = (s.empty() ? 0 : s.top());
    s.push(i);
  }
}

void solve_suf(int n) {
  stack<int> s;
  for (int i = n; i >= 1; --i) {
    while (!s.empty() && w[i]<=w[s.top()]) s.pop();
    suf[i] = (s.empty() ? n+1 : s.top());
    s.push(i);
  }
}

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

  solve_pre(n), solve_suf(n);
  for (int i = 1; i <= n; ++i) f[suf[i]-pre[i]-1] = max(f[suf[i]-pre[i]-1], w[i]);
  for (int i = n-1; i >= 1; --i) f[i] = max(f[i+1], f[i]);

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

CF366C Dima and Salad

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

using namespace std;

const int N = 100000, M = 100 + 5;
int a[M], b[M];
int p[2*N+10], f[2*N+10];

int main()
{
  int n, k; cin >> n >> k;
  for (int i = 1; i <= n; ++i) cin >> a[i];
  for (int i = 1; i <= n; ++i) cin >> b[i];
```

```
  memset(p, -1, sizeof(p)), memset(f, -1, sizeof(f));
  p[N] = f[N] = 0;

  for (int i = 1; i <= n; ++i) {
    int x = a[i] - k*b[i];
    for (int j = 1; j <= 2*N; ++j) {
      if (p[j] != -1) f[j+x] = max(f[j+x], p[j]+a[i]);
    }
    memcpy(p, f, sizeof(p));
  }

  cout << (f[N] ? f[N] : -1) << endl;
  return 0;
}
```

CF1288E Messenger Simulator

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

using namespace std;

const int N = 3e5 + 5, M = N<<1;

int tr[M];
int lowbit(int x) {
  return x & (-x);
}
void modify(int x, int k) {
  while (x < M) {
    tr[x] += k;
    x += lowbit(x);
  }
}
int query(int x) {
  int res = 0;
  while (x) {
    res += tr[x];
    x -= lowbit(x);
  }
  return res;
}

struct node {
  int minn, maxx;
} w[N];
int f[N];

int main()
{
    ios::sync_with_stdio(false);
```

```
    cin.tie(0);

    int n, m; cin >> n >> m;
    for (int i = 1; i <= n; ++i) {
      w[i] = {i, i};
      f[i] = N + i;
      modify(f[i], 1);
    }

    for (int i = 1; i <= m; ++i) {
      int x; cin >> x;
      w[x].minn = 1;
      w[x].maxx = max(w[x].maxx, query(f[x]));
      modify(f[x], -1);

      f[x] = N - i;
      modify(f[x], 1);
    }

    for (int i = 1; i <= n; ++i) {
      w[i].maxx = max(w[i].maxx, query(f[i]));
      cout << w[i].minn << " " << w[i].maxx << endl;
    }
    return 0;
}
```

CF1846G Rudolf and CodeVid-23

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

using namespace std;

const int maxn = 1024 + 5;
struct node {
  int dis, id;
  bool operator < (const node& p) const { return dis < p.dis; }
  bool operator > (const node& p) const { return dis > p.dis; }
};
vector<node> vec[maxn];
int p[maxn], a[maxn], b[maxn];

int strRead() {
  string s; cin >> s;
  int res = 0;
  for (char i : s) res = res*2 + (i-'0');
  return res;
}

int f[maxn];
bool st[maxn];
```

```
int dijkstra(int _start) {
  memset(f, -1, sizeof(f)); memset(st, false, sizeof(st));
  priority_queue<node, vector<node>, greater<node>>q;
  f[_start] = 0; q.push({f[_start], _start});

  while (!q.empty()) {
    node u = q.top(); q.pop();
    int d = u.dis, id = u.id;
    if (st[id]) continue;
    st[id] = true;
    for (node it : vec[id]) {
      if (f[it.id]==-1 || d+it.dis<f[it.id]) {
        f[it.id] = d + it.dis; q.push({f[it.id], it.id});
      }
    }
  }
  return f[0];
}

void solve() {
  int n, m; cin >> n >> m;
  for (int i = 0; i <= (1<<n)+2; ++i) vec[i].clear();

  int _start = strRead();
  for (int i = 1; i <= m; ++i) {
    cin >> p[i], a[i] = strRead(), b[i] = strRead();
  }

  int limit = (1<<n)-1;
  for (int i = 0; i <= limit; ++i) {
    for (int j = 1; j <= m; ++j) {
      int t = (i&(limit-a[j])) | b[j];
      vec[i].push_back({p[j], t});
    }
  }

//  cout << "---------------------- ";
  cout << dijkstra(_start) << endl;
}

int main()
{
  int T; cin >> T;
  while (T -- ) solve();
  return 0;
}
```

CF1512F Education

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

```cpp
using namespace std;

typedef long long LL;
const int maxn = 2e5 + 5;
int a[maxn], b[maxn];

int get_up(int a, int b) { return (a+b-1)/b; }

void solve() {
  int n, c; cin >> n >> c;
  for (int i = 1; i <= n; ++i) cin >> a[i];
  for (int i = 1; i <= n-1; ++i) cin >> b[i];

  LL res = 1e18;
  LL sum = 0, val = 0;
  for (int i = 1; i <= n; ++i) {
    res = min(res, sum+get_up(c-val, a[i]));
    if (i == n) continue;
    int t = get_up(b[i]-val, a[i]);
    sum += t+1, val += t*a[i] - b[i];
  }

//  cout << "---------------------- ";
  cout << res << endl;
}

int main()
{
  int T; cin >> T;
  while (T -- ) solve();
  return 0;
}
```