Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0307 – Мосолов Владислав Алексеевич

A	B	C	D	E	F	sum
100	75	100	35	20	49	379

Task A

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
#include <queue>
#include <random>
#include <set>
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair
```

```
using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

mt19937_64 rnd(157);

void release()
{
  cin.tie(0)->sync_with_stdio(0);
}

const i64 S = 82;
i64 dp[S];

int main()
{
#ifdef _DEBUG
  freopen("input.txt", "r", stdin);
#endif
  release();

  i64 a, b, n; cin >> a >> b >> n;

  dp[0] = 1;

  for (i64 x = 1; x <= n; ++x) {
    dp[x] = (x - a < 0 ? 0 : dp[x - a]) + (x - b < 0 ? 0 : dp[x - b]);
  }

  if (a > n) cout << 0;
  else if (b > n) cout << a * (dp[n - a]);
  else cout << a * (dp[n - a]) + b * (dp[n - b]);



}
```

Task B

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
```

```
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
#include <queue>
#include <random>
#include <set>
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair

using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

mt19937_64 rnd(157);

void release()
{
  cin.tie(0)->sync_with_stdio(0);
}

const i64 S = 82;
i64 dp[S];

i64 dist(i64 x, i64 y, i64 z, i64 x1, i64 y1, i64 z1) {
  i64 ans = 0;
  if (x < x1) ans += (x1 - x);
  if (y < y1) ans += (y1 - y);
  if (z < z1) ans += (z1 - z);
  return ans;
}

int main()
{
#ifdef _DEBUG
  freopen("input.txt", "r", stdin);
#endif
  release();

  i64 n, mn = INF; cin >> n;

  i64 x, y, z, x1, y1, z1; cin >> x >> y >> z >> x1 >> y1 >> z1;
```

```
  if (n <= 10) {
    for (i64 t = -n; t <= n; ++t) {
      for (i64 f = -n; f <= n; ++f) {
        for (i64 k = -n; k <= n; ++k) {
          if (t + f + k != 0) continue;
          mn = min(mn, dist(x, y, z, t, f, k) + dist(t, f, k, x1, y1, z1)
  );
          if (abs(t) == n) {
            mn = min(mn, dist(x, y, z, t, f, k) + dist(-t, -k, -f, x1, y1
  , z1));
          }
          if (abs(f) == n) {
            mn = min(mn, dist(x, y, z, t, f, k) + dist(-k, -f, -t, x1, y1
  , z1));
          }
          if (abs(k) == n) {
            mn = min(mn, dist(x, y, z, t, f, k) + dist(-f, -t, -k, x1, y1
  , z1));
          }
        }
      }
    }
    cout << mn;
    exit(0);
  }
  else {
    cout << dist(x, y, z, x1, y1, z1);
    exit(0);
  }
}
```

Task C

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
#include <queue>
#include <random>
#include <set>
```

```
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair

using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

mt19937_64 rnd(157);

void release()
{
  cin.tie(0)->sync_with_stdio(0);
}

const i64 S = 4e5;
i64 osn = 197, md = 1e9 + 7;
i64 pref[S];

i64 step(i64 a, i64 b) {
  if (b == 0) return 1;
  if (b == 1) return a;
  i64 k = step(a, b / 2);
  return ((k * k) % md * step(a, b % 2)) % md;
}

i64 mod(i64 a) {
  if (a > 0) return a % md;
  else {
    i64 k = abs(a) / md;
    a += k * md;
    if (a < 0) a += md;
    return a;
  }
}

i64 get(i64 l, i64 r) {
  if (l == 0) return pref[r];
  return mod(pref[r] - step(osn, r - l + 1) * pref[l - 1]);
}

int main()
{
#ifdef _DEBUG
```

```
  freopen("input.txt", "r", stdin);
#endif
  release();

  string num; cin >> num;

  i64 n, n1; string k;
  cin >> n >> k;
  n1 = n;

  if (num == "first") {
    vector <i64> len;
    for (i64 x = 1; x <= 8; ++x) {
      if (n1 == 0) break;
      len.push_back(min((unsigned long long)n1, rnd() % (max(i64(1), n /
    10)) + (i64)1));
      n1 -= len.back();
    }
    if (n1 != 0) len.push_back(n1);
    i64 pos = 0;
    for (auto x : len) {
      i64 r = pos + x - 1, hsh = 0;
      for (i64 ind = pos; ind <= r; ++ind) {
        hsh = (hsh * osn + k[ind]) % md;
      }
      pos = r + 1;
      cout << hsh << "_";
    }
  }
  else {
    vector <i64> len;
    for (i64 x = 1; x <= 8; ++x) {
      if (n1 == 0) break;
      len.push_back(min((unsigned long long)n1, rnd() % (max(i64(1), n /
    10)) + (i64)1));
      n1 -= len.back();
    }
    if (n1 != 0) len.push_back(n1);
    string ciph; cin >> ciph;
    vector <i64> key;
    i64 val = 0;
    for (i64 x = 0; x < ciph.size(); ++x) {
      if (ciph[x] >= '0' && ciph[x] <= '9') val = (val * 10 + (ciph[x] -
    '0'));
      else {
        key.push_back(val);
        val = 0;
      }
    }
    k += k;
    i64 hsh = 0;
    for (i64 x = 0; x < k.size(); ++x) {
      hsh = (hsh * osn + k[x]) % md;
```

```
      pref[x] = hsh;
    }
    for (i64 x = 0; x < n; ++x) {
      i64 pos = x, ind = 0;
      bool ok = true;
      for (auto y : len) {
        i64 r = pos + y - 1;
        if (get(pos, r) == key[ind]) {
          ++ind;
          pos = r + 1;
        }
        else {
          ok = false;
        }
      }
      if (ok) {
        cout << "YES"; exit(0);
      }
    }
    cout << "NO";
  }
}
```

Task D

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
#include <queue>
#include <random>
#include <set>
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair
```

```
using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

mt19937 rnd(996);

void release()
{
  cin.tie(0)->sync_with_stdio(0);
}

int main()
{
/*#ifdef _DEBUG
  freopen("input.txt", "r", stdin);
#endif*/

  release();

  i64 n; cin >> n;

  while (true) {
    i64 k = rnd() % n + 1, k1 = rnd() % n + 1;
    cout << k << " " << k1 << endl;
    char f; cin >> f;
    if (f == '!') {
      cout << endl; exit(0);
    }
  }
}
```

Task E

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
```

```
#include <queue>
#include <random>
#include <set>
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair

using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

void release()
{
  cin.tie(0)->sync_with_stdio(0);
}

const i64 S = 1e5 + 10;
vector <i64> p[S];
i64 ans[S], t[S]; bool nd = false;

void dfs(i64 v, i64 num) {
  if (num >= (i64)S - 5 || nd) {
    nd = true;
    return;
  }
  ans[num] = v;
  ++t[v];
  dfs(p[v][(t[v] - 1) % (i64(p[v].size()))], num + 1);
}

int main()
{
#ifdef _DEBUG
  freopen("input.txt", "r", stdin);
#endif
  release();

  i64 n, q; cin >> n >> q;

  for (i64 x = 1; x <= n; ++x) {
    i64 k; cin >> k;

    for (i64 y = 1; y <= k; ++y) {
      i64 d; cin >> d;
```

```
      p[x].push_back(d);
    }
  }

  dfs(1, 0);

  for (i64 x = 1; x <= q; ++x) {
    i64 quer; cin >> quer;
    cout << ans[quer] << " ";
  }
}
```

Task F

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <stdio.h>
#include <stdint.h>
#include <algorithm>
#include <bitset>
#include <chrono>
#include <ctime>
#include <thread>
#include <deque>
#include <fstream>
#include <iomanip>
#include <iostream>
#include <assert.h>
#include <iterator>
#include <map>
#include <queue>
#include <random>
#include <set>
#include <assert.h>
#include <stack>
#include <string>
#include <tuple>
#include <cstdint>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#define mp make_pair

using namespace std;
typedef int i32;
typedef long long i64;
typedef long double ld;

const i64 INF = 1e18;

void release()
{
  cin.tie(0)->sync_with_stdio(0);
```

```
}

const i64 S = 52;
i64 p[S][S];

int main()
{
#ifdef _DEBUG
  freopen("input.txt", "r", stdin);
#endif

  release();

  i64 a, b; cin >> a >> b;

  for (i64 x = 1; x <= a; ++x) {
    for (i64 y = 1; y <= b; ++y) {
      char k; cin >> k;
      p[x][y] = k - '0';
    }
  }

  i64 ans = 0, ans1 = 0;

  for (i64 x = 1; x <= a; x += 2) {
    for (i64 y = 1; y <= b; ++y) {
      ans += (p[x][y] * p[x + 1][y]);
    }
  }

  for (i64 x = 1; x <= a; ++x) {
    for (i64 y = 1; y <= b; y += 2) {
      ans1 += (p[x][y] * p[x][y + 1]);
    }
  }

  if (ans > ans1) {

    cout << ans << endl;

    for (i64 x = 1; x <= a; ++x) {
      for (i64 y = 1; y <= b; ++y) {
        if (x % 2 == 1) cout << 'T';
        else cout << 'B';
      }
      cout << endl;
    }
  }
  else {
    cout << ans1 << endl;

    for (i64 x = 1; x <= a; ++x) {
      for (i64 y = 1; y <= b; ++y) {
```

```
        if (y % 2 == 1) cout << 'L';
        else cout << 'R';
      }
      cout << endl;
    }
  }
}
```