Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	80	58	0	438

Task A ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << "␣" << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);

  int n;
  cin >> n;
  vector<pair<int, int> > vc(n);
  for (int i = 0; i < n; ++i) {
    int a, b;
    cin >> a >> b;
    while (a % 10 == 0) {
      ++b;
      a /= 10;
    }
    vc[i] = {b, a};
  }
  sort(vc.begin(), vc.end());
  int res = vc[0].first;
  ll sum = vc[0].second;
  for (int i = 1; i < n; ++i) {
    int nxt = vc[i].first;
    while (nxt > vc[i-1].first && sum % 10 == 0) {
      --nxt;
      ++res;
      sum /= 10;
    }
    if (nxt != vc[i-1].first)
      break;
    sum += vc[i].second;
  }
  while (sum % 10 == 0) {
    sum /= 10;
    ++res;
  }
  cout << res << endl;

  return 0;
}
```

Task B ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << "␣" << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

const string instructions[] = {
  string("Wait"),
  string("Flip␣and␣wait"),
  string("Stop")
};
int q(int instr) {
  cout << instructions[instr] << endl;
  if (instr == 2) return 0;
  string ans;
  cin >> ans;
  if (ans[0] != 'B' || ans[1] != 'e')
    exit(0);
  int cnt = 0;
  for (int i = 1; i < ans.size(); i += 2) {
    if (ans[i] != 'e') break;
    ++cnt;
  }
  return cnt;
}

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);

  int n;
  cin >> n;
  vector<int> a(n);
  for (int i = 0; i < n; ++i)
    cin >> a[i];
  for (int i = n-1; i >= 0; --i) {
    for (int j = 0; j < a[i]; ++j) {
      int unt = i + 1;
      int ans = q(1);
      while (unt > 0) {
        unt -= ans;
        if (unt > 0)
```

```
                ans = q(0);
            }
        }
    }
    q(2);

    return 0;
}
```

Task C ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << "␣" << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

map<string, string> enc, decd;
void setupFor(int n) {
  for (int i = 0; i < (1 << n); ++i) {
    bool ok = false;
    for (int j = 0; j < (1 << n); ++j) {
      int cnt = 0;
      for (int k = 0; k < n; ++k) {
        if ((j >> k) & 1) ++cnt;
      }
      if (cnt < n/2) continue;
      string s; s.resize(n);
      string encS; encS.resize(n);
      for (int k = 0; k < n; ++k) {
        if ((j >> k) & 1) encS[k] = '?';
        else if ((i >> k) & 1) encS[k] = '1';
        else encS[k] = '0';
        if ((i >> k) & 1) s[k] = '1';
        else s[k] = '0';
      }
      if (!decd.count(encS)) {
        enc[s] = encS;
        decd[encS] = s;
        ok = true;
        break;
      }
    }
    assert(ok);
  }
}

string encodeSmall(string & s) {
  return enc[s];
}

string decodeSmall(string & s) {
  return decd[s];
}

const int amountOfChecks = 20;
const int N = 100;
int checkPositions[N];
string encodeLarge(string & s) {
  int cnt[2] = {0,0};
  assert(s.size() >= 100);
  int n = s.size();
  for (int i = 0; i < s.size(); ++i) {
    ++cnt[s[i] == '1'];
  }
  char c = '0';
  if (cnt[1] > cnt[0]) c = '1';
  string res;
  for (int i = 0; i < n; ++i) {
    if (s[i] == c)
      res.push_back('?');
    else
      res.push_back(s[i]);
  }
  if (cnt[0] == 0 || cnt[1] == 0) {
    for (int i = 0; i < n; ++i) {
      if (i < 100 && checkPositions[i])
        res[i] = c;
    }
  }
  return res;
}

string decodeLarge(string & s) {
  assert(s.size() >= 100);
  char c = '0';
  int n = s.size();
  int cnt = 0;
  for (int i = 0; i < n; ++i) {
    if (s[i] != '?' && s[i] != c) {
      assert(!cnt);
      c = s[i];
    }
    if (s[i] == c) ++cnt;
  }
  string res;
  bool controlPositionsPass = cnt == 20;
  for (int i = 0; i < n && controlPositionsPass; ++i) {
    if (i < 100 && checkPositions[i] && s[i] != c) controlPositionsPass = false;
    if ((i >= 100 || !checkPositions[i]) && s[i] != '?') controlPositionsPass = false;
  }
  if (controlPositionsPass) {
    for (int i = 0; i < n; ++i) {
      res.push_back(c);
    }
  } else {
    char ch = (c == '0' ? '1' : '0');
    for (int i = 0; i < n; ++i) {
      if (s[i] == '?')
        res.push_back(ch);
      else
        res.push_back(c);
    }
  }
  return res;
}

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);

  fill(checkPositions, checkPositions + N, 0);
  rng.seed(1810);
  for (int i = 0; i < 20; ++i) {
    int ind = uid(rng) % 100;
    if (checkPositions[ind]) --i;
    else checkPositions[ind] = 1;
  }
  for (int i = 2; i <= 10; ++i)
    setupFor(i);
  int n;
  cin >> n;
  for (int i = 0; i < n; ++i) {
    string s;
    cin >> s;
    bool encode = true;
    for (int j = 0; j < s.size(); ++j) {
      encode &= s[j] != '?';
    }
    if (s.size() <= 10) {
      if (encode)
        cout << encodeSmall(s) << endl;
      else
        cout << decodeSmall(s) << endl;
    } else {
      if (encode)
        cout << encodeLarge(s) << endl;
      else
        cout << decodeLarge(s) << endl;
    }
  }

  return 0;
}
```

Task D ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << "␣" << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

const int MOD = 998244353;
int add(int a, int b) {
  return a + b >= MOD ? a + b - MOD : a + b;
}
int sub(int a, int b) {
  return a - b < 0 ? a - b + MOD : a - b;
}
int mul(int a, int b) {
  return 1ll * a * b % MOD;
}

const int BASE[10] = { 1,
  //100000000
  808258749,
  //200000000
    117153405,
  //300000000
    761699708,
  //400000000
    573994984,
  //500000000
    62402409,
  //600000000
    511621808,
  //700000000
    242726978,
  //800000000
  887890124,
  //900000000
  875880304
};

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);
  int n;
  cin >> n;
  ++n;
  int cut = 1e8;
  ll f = BASE[int(n / cut)];
  for (int i = n / cut * cut + 1; i <= n; ++i) {
    f *= i;
    f %= MOD;
  }
  cout << 0 << '␣' << f << endl;

  return 0;
}
```

Task E ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << " " << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);

  int n;
  cin >> n;
  vector<int> v(n);
  for (int i = 0; i < n; ++i)
    cin >> v[i];
  int q;
  cin >> q;
  while (q--) {
    int a, b, d;
    cin >> a >> b >> d;
    ll cnt = 0;
    for (int i = a; i < b; ++i) {
      cnt += (d + v[i] - 1) / v[i];
    }
    cout << cnt << endl;
  }

  return 0;
}
```

Task F ()

```
#include <iostream>
#include <utility>
#include <vector>
#include <map>
#include <set>
#include <algorithm>
#include <math.h>
#include <numeric>
#include <queue>
#include <deque>
#include <functional>
#include <string>
#include <random>
#include <bitset>
#include <cassert>
#include <iomanip>
#include <chrono>
#include <random>
using namespace std;
typedef long long ll;
typedef long double ld;
inline void smax(int&a,int b){if(b>a)a=b;}
inline void smin(int&a,int b){if(b<a)a=b;}
inline void smax(ll&a,ll b){if(b>a)a=b;}
inline void smin(ll&a,ll b){if(b<a)a=b;}
const ll LLINF = ll(2e18) + 13; const int INF = int(1e9) + 7; mt19937 rng((unsigned int)(time(
    nullptr) + 'm' + 'a' + 'n' + 'u' + 'l'));
uniform_int_distribution<int> uid(0, INF);
#ifdef SSLOCAL
void debug_out() { cerr << endl; }
template <typename Head, typename... Tail> void debug_out(Head H, Tail... T) { cerr << "␣" << H;
    debug_out(T...);}
#define debug(...) cerr << "[" << #__VA_ARGS__ << "]:", debug_out(__VA_ARGS__)
#else
#define debug(...)
#endif

const ll N = 1ll << 51;
const ll OFFSET = 1ll << 50;
struct Node {
  Node *l, *r;
  int val, push;
  Node(): val(0), push(0), l(nullptr), r(nullptr) {}
};
Node *root = new Node();
void _push(Node *v, ll vl, ll vr) {
  if (v->push) {
    v->val += v->push;
    if (vr - vl > 1) {
      if (!v->l) v->l = new Node();
      v->l->push += v->push;
      if (!v->r) v->r = new Node();
      v->r->push += v->push;
    }
    v->push = 0;
  }
}
int get() {
  _push(root, 0, N);
  return root->val;
}
void add(ll l, ll r, Node *v=root, ll vl=0, ll vr=N) {
  _push(v, vl, vr);
  if (l >= vr || vl >= r) return;
  if (l <= vl && vr <= r) {
    v->push = 1;
    _push(v, vl, vr);
    return;
  }
  ll vm = vl + vr >> 1;
  if (!v->l) v->l = new Node();
  add(l, r, v->l, vl, vm);
  if (!v->r) v->r = new Node();
  add(l, r, v->r, vm, vr);
  v->val = max(v->l->val, v->r->val);
}

int32_t main() {
  ios_base::sync_with_stdio(0); cin.tie(0);

  int n;
  cin >> n;
  ll pl = 0, pr = 0;
  add(OFFSET, OFFSET + 1);
  for (int i = 0; i < n; ++i) {
    int l, r;
    cin >> l >> r;
    pl += l;
    pr += r;
    add(pl + OFFSET, pr + OFFSET);
  }
  cout << get() << endl;

  return 0;
}
```