The event of zero and one
ACM Template

HENAN UNIVERSITY

1912

目录

1 String **3**
1.1 String Hash 3
1.2 Manacher 3

2 Math **4**
2.1 Linear Sieve 4
2.2 组合数预处理 5

3 Data Structure **5**
3.1 Fenwick 5
3.2 K-Dtree 6
3.3 可持久化平衡树 7

4 Others **8**
4.1 Debugger 8
4.2 Fast IO 9

1 String

1.1 String Hash

```
302f #include <bits/stdc++.h>
421c using namespace std;
4085 typedef long long ll;
0dfa #ifdef LOCAL
9e47 #include <debugger>
a8cb #else
8bbb #define debug(...) 42
1937 #endif
8048 template <typename T> void chkmax(T &x, T y) { x = x >= y ? x : y; }
d0e2 template <typename T> void chkmin(T &x, T y) { x = x <= y ? x : y; }
c429 mt19937_64 rnd(chrono::steady_clock::now().time_since_epoch().count());
8f98 constexpr int N = 1000001;
1948 using ull = unsigned long long;
e2fd using HH = pair<unsigned long long, unsigned long long> ;
b301 string s;
57ea HH base, Hash[N + 1];
9776 constexpr HH mod = {1000000009, 1000000007};
4d27 HH operator* (const HH &a, const HH &b) {
cea9   return {(a.first * b.first) % mod.first,
9c20     (a.second*b.second)%mod.second};
95cf }
e1b8 HH operator+ (const HH &a, const HH &b) {
7223   return {(a.first + b.first) % mod.first,
8cae     (a.second + b.second) % mod.second};
95cf }
9490 HH operator- (const HH &a, const HH &b) {
d4c8   return {(a.first - b.first + mod.first) % mod.first,
0c13     (a.second - b.second + mod.second) % mod.second
329b   };
427e //return{((a.first%mod.first)+mod.first-(b.first%mod.first))%mod.first,
427e //   ((a.second%mod.second)+mod.second-(b.second%mod.second))%mod.second};
95cf }
cf66 HH operator* (const HH &a, const ull &b) {
01ee   return {(a.first * b) % mod.first, (a.second * b) % mod.second};
95cf }
3117 int main() {
7618   ios::sync_with_stdio(false);
498a   cin.tie(nullptr);
04cb   base = {rnd() % mod.first, rnd() % mod.second};
```

```
  Hash[0] = {1, 1};                                                      4b64
  for(int i = 1; i <= N; i ++ ) Hash[i] = Hash[i - 1] * base ;           8e29
  int n, q; cin >> n >> q >> s;                                          96ec
                                                                         427e
  vector<HH> a(n + 1), b(n + 1);                                         4145
  a[0] = b[0] = {1, 1};                                                  5998
  for(int i = 1; i <= n; i ++ ) {                                        6dbf
    a[i] = a[i - 1] * base + a[0] *  (s[i - 1] - 'a' + 1) ;              edbb
  }                                                                      95cf
  reverse(s.begin(), s.end());                                           bf92
                                                                         427e
  for(int i = 1; i <= n; i ++ ) {                                        6dbf
    b[i] = b[i - 1] * base + b[0] *  (s[i - 1] - 'a' + 1);               e6c2
  }                                                                      95cf
  auto get1 = [&] (int l, int r) {                                       f03f
    return a[r] - (a[l - 1] * Hash[r - l + 1]);                          ab6a
  };                                                                     329b
                                                                         427e
  auto get2 = [&] (int l, int r) {                                       05b4
    return b[r] - (b[l - 1] * Hash[r - l + 1]);                          29e3
  };                                                                     329b
  while(q -- ) {                                                         2cc8
    int l, r; cin >> l >> r;                                             c439
    int x = r - l + 1;                                                   320d
    bool flag = (get1(l, r) == get2(n - l - x + 2, n - l + 1));          096a
    if(flag) cout << "Budada\n";                                         ff4e
    else{                                                                037f
      if(x & 1) cout << "Putata\n";                                      8ecd
      else cout << "Budada\n";                                           8951
    }                                                                    95cf
  }                                                                      95cf
  return 0;                                                              7021
}                                                                        95cf
```

1.2 Manacher

```
// https://www.acwing.com/problem/content/description/141/               427e
#include <bits/stdc++.h>                                                 302f
using namespace std;                                                     421c
                                                                         427e
constexpr int N = 2000010;                                               3596
                                                                         427e
```

```
3d88 int n, m, Case;
adbc char s[N], str[N];
b8fe int p[N];
427e
d9ed int manacher() {
44e9   int rt = 0, mid = 0;
5839   int res = 0;
e052   for(int i = 1; i <= m; i ++ ) {
05a8     p[i] = i < rt ? min(p[2 * mid - i], rt - i) : 1;
9ad1     while(str[i + p[i]] == str[i - p[i]]) ++ p[i];
9e84     if(i + p[i] > rt) {
5400       rt = i + p[i];
c220       mid = i;
95cf     }
cbe1     res = max(res, p[i] - 1);
95cf   }
244d   return res;
95cf }
427e
3117 int main() {
d703   str[0] = '!', str[1] = '#';
cf96   while(scanf("%s", s), s[0] != 'E') {
5264     n = strlen(s);
1294     for(int i = 0; i < n; i ++ ) {
e0c8       str[i * 2 + 2] = s[i];
c0d0       str[i * 2 + 3] = '#';
95cf     }
5256     m = n * 2 + 1;
0a17     str[m + 1] = '@';
1156     printf("Case␣%d:␣%d\n", ++ Case, manacher());
95cf   }
7021   return 0;
95cf }
```

2 Math

2.1 Linear Sieve

```
302f #include<bits/stdc++.h>
421c using namespace std;
68e4 const int maxn = 1e7+10;
4085 typedef long long ll;
727f bool used[maxn];
efe5 int mu[maxn];
7c8f vector<int> prime;
c882 ll f[maxn];
a0b1 int low[maxn];
22c5 void sieve(int size){
427e     //f:multiplicative function;
7d97     assert(size < maxn);
7f5a     mu[1] = 1;
c6b9     f[1] = 1;
40bd     for (int i=2;i<=size;i++){
efb1         if (!used[i]){
1024             prime.push_back(i);
7171             mu[i] = -1;
427e             //f:TODO
c21b             low[i] = i;
95cf         }
eb1a         for (int j = 0;j < prime.size();j++){
d3c2             ll nxt = 1ll * i * prime[j];
b561             if (nxt > size)break;
6b89             used[nxt] = 1;
073a             if (i % prime[j]){
b9b8                 low[nxt] = prime[j];
66f9                 mu[nxt] = -mu[i];
427e                 //f: mod or not?
7225                 f[nxt] = f[i] * f[prime[j]];
8e2e             }else{
734b                 low[nxt] = prime[j] * low[i];
8ec3                 mu[nxt] = 0;
b401                 if (low[nxt] != nxt){
427e                     //mod or not?
4d18                     f[nxt] = 1ll * f[low[nxt]] * f[nxt/low[nxt]];
8e2e                 }else{
427e                     // i = prime[j] ^ k
427e                     //f:TODO
95cf                 }
6173                 break;
95cf             }
95cf         }
95cf     }
95cf }
3117 int main(){
ff91     sieve(1e7);
7021     return 0;
} 95cf
```

2.2 组合数预处理

```
const int MOD = 998244353; 5bf2
 427e
inline int mod(int x) {return x >= MOD ? x - MOD : x;} 182f
 427e
inline int ksm(int a, int b) { 7ce8
  int ret = 1; a = mod(a); 3bd1
  for(; b; b >>= 1, a = 1LL * a * a % MOD) if(b & 1) ret = 1LL * ret * a % MOD; f1be
  return ret; ee0f
} 95cf
 427e
template<int MOD> 8f3b
struct modint { 9ba1
  int x; 3c9e
  modint() {x = 0; } 187e
  modint(int y) {x = y;} 5ada
  inline modint inv() const { return modint{ksm(x, MOD - 2)}; } 478f
  explicit inline operator int() { return x; } feef
  friend inline modint operator + (const modint &a, const modint& b) { return modint(mod(a.x + b.x)); } d916
  friend inline modint operator - (const modint &a, const modint& b) { return modint(mod(a.x - b.x + MOD)); } 35f7
  friend inline modint operator * (const modint &a, const modint& b) { return modint(1ll * a.x * b.x % MOD); } 8c7e
  friend inline modint operator / (const modint &a, const modint& b) { return modint(1ll * a.x * b.inv().x % MOD); } beed
  friend inline modint operator - (const modint &a) { return modint(mod(MOD - a.x)); } 5fa9
  friend inline modint& operator += (modint &a, const modint& b) { return a = a + b; } 1b61
  friend inline modint& operator -= (modint &a, const modint& b) { return a = a - b; } aa0b
  friend inline modint& operator *= (modint &a, const modint& b) { return a = a * b; } 5db7
  friend inline modint& operator /= (modint &a, const modint& b) { return a = a / b; } eadc
  friend auto &operator >> (istream &i, modint &a) {return i >> a.x; } ef3a
  friend auto &operator << (ostream &o, const modint &z) { return o << z.x; } c8b8
  inline bool operator == (const modint &b) { return x == b.x; } 1156
  inline bool operator != (const modint &b) { return x != b.x; } ae48
  inline bool operator < (const modint &a) { return x < a.x; } e758
  inline bool operator <= (const modint &a) { return x <= a.x; } a92c
  inline bool operator > (const modint &a) { return x > a.x; } 51b4
  inline bool operator >= (const modint &a) { return x >= a.x; } d380
 427e
}; 329b
 427e
typedef modint<MOD> mint; d6e9
 427e
inline mint ksm(mint a, int b) { 75c9
  mint ret = 1; 80f5
  for(; b; b >>= 1, a = a * a ) if(b & 1) ret = ret * a ; 7860
  return ret; ee0f
} 95cf
 427e
const int N = 2e5 + 10; 8334
 427e
mint fact[N + 1], infact[N + 1], inv[N + 1]; ed5b
 427e
void init() { 5d53
  fact[0] = 1; for(int i = 1; i <= N; ++ i ) { fact[i] = fact[i - 1] * i; } 8ae2
  infact[N] = ksm(fact[N], MOD - 2); for(int i = N - 1; i >= 0; -- i ) infact[i] = infact[i + 1] * (i + 1); 4e8b
  inv[0] = inv[1] = 1; for(int i = 2; i <= N; ++ i) inv[i] = inv[MOD % i] * (MOD - MOD / i); ea9a
} 95cf
```

3 Data Structure

3.1 Fenwick

```
template <typename T> b7ec
class fenwick { c881
 public: 63d4
  vector<T> fenw; 2d6b
  int n; 5c83
 427e
  fenwick(int _n) : n(_n) { 2372
    fenw.resize(n); bf4d
  } 95cf
 427e
```

```cpp
  void modify(int x, T v) {
    while (x < n) {
      fenw[x] += v;
      x |= (x + 1);
    }
  }

  T get(int x) {
    T v{};
    while (x >= 0) {
      v += fenw[x];
      x = (x & (x + 1)) - 1;
    }
    return v;
  }
};
```

3.2 K-Dtree

```cpp
/*---KD-Tree---*/
template<class T>
inline T sqr(T&x){return x*x;}
namespace KDT {
    const int N = 1e5 + 10, M = N << 1;
#define lc ch[x][0]
#define rc ch[x][1]
  struct pnt {
    int x[2];
  }a[N];
  int opt;
  struct node {
    int dis;
    node(int a = 0) {
      dis = a;
    }
    inline bool operator <(const node& other)const { return dis > other.dis; }
  };//???
  priority_queue<node>q;
  int ans;
  inline bool cmp(register pnt a, register pnt b) { return a.x[opt] < b.x[opt]; }
inline int f(pnt a, pnt b) { return sqr(a.x[0] - b.x[0]) + sqr(a.x[1] - b.x[1]); }
int ch[M][2], minn[M][2], maxn[M][2], cnt;
pnt*d[M];
inline void pushup(const int&x) {
  for (int i = 0; i < 2; ++i)
    maxn[x][i] = minn[x][i] = d[x]->x[i];
  if (lc)
    for (int i = 0; i < 2; ++i)
      maxn[x][i] = max(maxn[x][i], maxn[lc][i]),
      minn[x][i] = min(minn[x][i], minn[lc][i]);
  if (rc)
    for (int i = 0; i < 2; ++i)
      maxn[x][i] = max(maxn[x][i], maxn[rc][i]),
      minn[x][i] = min(minn[x][i], minn[rc][i]);
}inline int build(int L, int R, int k) {
  if (L > R)return 0;
  int x = ++cnt;
  opt = k;
  int mid = (L + R) >> 1;
  nth_element(a + L, a + mid, a + R + 1, cmp);
  d[x] = &a[mid];
  lc = build(L, mid - 1, k ^ 1), rc = build(mid + 1, R, k ^ 1);
  pushup(x);
}inline int mindis(pnt&a, int b) {
  return sqr(max(a.x[0]-maxn[b][0],0)+max(minn[b][0]-a.x[0],0))+sqr(max(a.x[1]-maxn[b][1],0)+max(minn[b][1]-a.x[1],0));
}inline int maxdis(pnt&a, int b) {//????
  return max(sqr(maxn[b][0] - a.x[0]), sqr(minn[b][0] - a.x[0])) + max(sqr(maxn[b][1] - a.x[1]), sqr(minn[b][1] - a.x[1]));
 }inline void kqueryMax(int x, pnt&O) {//k?????
  if(!x)return;
  int dl = -inf, dr = -inf;
  if (lc)dl = maxdis(O, lc);
    if (rc)dr = maxdis(O, rc);
    if (d[x] != &O) {
      int dis = f(*d[x], O);
      if (q.top().dis < dis)
        q.pop(), q.push(dis);
    }
    if (dl > dr) {
      if (q.top().dis < dl)kqueryMax(lc, O);
      if (q.top().dis < dr)kqueryMax(rc, O);
    }
037f       else {
1330         if (q.top().dis < dr)kqueryMax(rc, O);
b055         if (q.top().dis < dl)kqueryMax(lc, O);
95cf       }
b10d    }inline void queryMin(int x, pnt&O) {//????
d2c9     if (!x)return;
68dd     int dl = inf, dr = inf;
3938     if (lc)dl = mindis(O, lc);
703c     if (rc)dr = mindis(O, rc);
34b9     if (d[x] != &O)//??????
2c2d        ans = min(ans, f(*d[x], O));
f705     if (dl < dr) {
d296       if (dl < ans)queryMin(lc, O);
0788       if (dr < ans)queryMin(rc, O);
95cf     }
037f     else {
0788       if (dr < ans)queryMin(rc, O);
d296       if (dl < ans)queryMin(lc, O);
95cf     }
28f2    }inline void queryMax(int x, pnt& O) {//????
d2c9     if (!x)return;
6060     int dl = -inf, dr = -inf;
6592     if (lc)dl = maxdis(O, lc);
b6db     if (rc)dr = maxdis(O, rc);
34b9     if (d[x] != &O)//??????
71fd        ans = max(ans, f(*d[x], O));
d23e     if (dl > dr) {
d809       if (ans < dl)queryMax(lc, O);
5162       if (ans < dr)queryMax(rc, O);
95cf     }
037f     else {
5162       if (ans < dr)queryMax(rc, O);
d809       if (ans < dl)queryMax(lc, O);
95cf     }
95cf   }
dfd3 #undef lc
2795 #undef rc
95cf }
```

3.3 可持久化平衡树

```
427e // code by Sansi
// luogu p3835                                                    427e
#include <bits/stdc++.h>                                          302f
using namespace std;                                              421c
const int N = 5e5 + 10;                                           115a
int rt[N], cnt;                                                   0284
std::mt19937 rnd(233);                                            4674
struct node{                                                      f1a7
  int l, r, val, key, size;                                       bf79
}tr[N * 50];                                                      7a7d
int new_node(int val){                                            7322
  tr[++cnt].val = val;                                            ee23
  tr[cnt].key = rnd();                                            2740
  tr[cnt].size = 1;                                               330f
  return cnt;                                                     6808
}                                                                 95cf
void pushup(int u){                                               949a
  tr[u].size = tr[tr[u].l].size + tr[tr[u].r].size + 1;           8893
}                                                                 95cf
void split(int u, int val, int& x, int& y){                       8284
  if(!u){                                                         0ef5
    x = y = 0; return;                                            89aa
  }                                                               95cf
  if(tr[u].val <= val){                                           d821
    x = ++ cnt; tr[x] = tr[u];                                    8d05
    split(tr[x].r, val, tr[x].r, y);                              f7fc
    pushup(x);                                                    9a13
  }                                                               95cf
  else{                                                           037f
    y = ++ cnt; tr[y] = tr[u];                                    391b
    split(tr[y].l, val, x, tr[y].l);                              127e
    pushup(y);                                                    3623
  }                                                               95cf
}                                                                 95cf
int merge(int x, int y){                                          41b9
  if(!x || !y) return x + y;                                      caae
  if(tr[x].key > tr[y].key){                                      e60c
    int p = ++ cnt; tr[p] = tr[x];                                8029
    tr[p].r = merge(tr[p].r, y);                                  848d
    pushup(p); return p;                                          fa64
  }                                                               95cf
  else{                                                           037f
    int p = ++ cnt; tr[p] = tr[y];                                2299
    tr[p].l = merge(x, tr[p].l);                                  4a88
    pushup(p); return p;                                          fa64
95cf   }
95cf }
4cb9 void insert(int& root, int val){
2b8d   int x, y, z;
6564   split(root, val, x, y);
947d   z = new_node(val);
5597   root = merge(merge(x, z), y);
95cf }
afff void Delete(int& root, int val){
2b8d   int x, y, z;
4008   split(root, val, x, z);
b921   split(x, val - 1, x, y);
774d   y = merge(tr[y].l, tr[y].r);
6df0   root = merge(merge(x, y), z);
95cf }
5e8d int Rank(int& root, int val){
0f8b   int x, y;
b88e   split(root, val - 1, x, y);
5945   int res = tr[x].size + 1;
29be   root = merge(x, y);
244d   return res;
95cf }
6b4c int K_th(int& root, int k){
f9c3   int u = root;
7ce6   while(u){
3ee0     if(tr[tr[u].l].size + 1 == k)
6173       break;
0390     else if(tr[tr[u].l].size >= k)
e8da       u = tr[u].l;
037f     else{
f635       k -= tr[tr[u].l].size + 1;
59d3       u = tr[u].r;
95cf     }
95cf   }
e099   return tr[u].val;
95cf }
769b int pre(int& root, int val){
2452   int x, y, k, res;
b88e   split(root, val - 1, x, y);
b9ba   if(!x) return -2147483647;
271d   k = tr[x].size;
18fe   res = K_th(x, k);
29be   root = merge(x, y);
244d   return res;
```

```
}                                                 95cf
int nex(int& root, int val){                      22b3
  int x, y, res;                                  3627
  split(root, val, x, y);                         6564
  if(!y) return 2147483647;                       0d6c
  else res = K_th(y, 1);                          0b2c
  root = merge(x, y);                             29be
  return res;                                     244d
}                                                 95cf
void solve(){                                     9627
  int n; cin >> n;                                b9aa
    for(int i = 1; i <= n; i ++){                 6dbf
        int t, op, x; cin >> t >> op >> x;        099d
        rt[i] = rt[t];                            dc05
        if(op == 1) insert(rt[i], x);             9a25
        if(op == 2) Delete(rt[i], x);             797d
        if(op == 3) cout << Rank(rt[t], x) << '\n';   1bb2
        if(op == 4) cout << K_th(rt[t], x) << '\n';   5fc8
        if(op == 5) cout << pre(rt[t], x) << '\n';    ac30
        if(op == 6) cout << nex(rt[t], x) << '\n';    97df
    }                                             95cf
}                                                 95cf
int main(){                                       3117
    ios::sync_with_stdio(false);                  7618
    cin.tie(nullptr);                             498a
    solve();                                      ccd1
    return 0;                                     7021
}                                                 95cf
```

4 Others

4.1 Debugger

```
namespace debugger {                              e71a
#ifdef DEBUG                                      d966
    template <typename T>                         b7ec
    void __print_var(string_view name, const T & x) { std::cerr << name << "␣=␣"   d71e
       << x; }
    void __print_var(string_view name, const string & x) { std::cerr << name <<   e2f6
      "␣=␣\"" << x << "\""; }
    void __print_var(string_view name, const char & x) { std::cerr << name << "␣   d057
      =␣\'" << x << "\'"; }
```

```
b7ec      template <typename T>
9e23      void __print_var(string_view name, const vector<T>& x) {
a881          std::cerr << name << "␣=␣";
5efd          bool is_first = true;
3768          for (auto & ele : x) std::cerr << (is_first ? (is_first = false, "[") :
                ",␣") << ele;
c731          std::cerr << "]";
95cf      }
b7ec      template <typename T>
7dcc      void __print_var(string_view name, const set<T>& x) {
a881          std::cerr << name << "␣=␣";
5efd          bool is_first = true;
0be6          for (auto & ele : x) std::cerr << (is_first ? (is_first = false, "{") :
                ",␣") << ele;
c8e5          std::cerr << "}";
95cf      }
18ba      template <typename K, typename V>
b1c0      void __print_var(string_view name, const map<K, V>& x) {
a881          std::cerr << name << "␣=␣";
5efd          bool is_first = true;
8150          for (auto & [k, v] : x) std::cerr << (is_first ? (is_first = false, "{")
                 : ",␣") << "(" << k << ":␣" << v << ")";
c8e5          std::cerr << "}";
95cf      }
b7ec      template <typename T>
e9e6      void __log(string_view name, const T & x) {
514d          __print_var(name, x); std::cerr << '\n';
95cf      }
9dda      template <typename T, typename... Ts>
fa46      void __log(string_view name, const T & x, const Ts&... others) {
064b          size_t pos = name.find(',');
c2b2          __print_var(name.substr(0, pos), x); std::cerr << ",␣";
5312          __log(name.substr(pos + 1), others...);
95cf      }
427e
65f4  #define LOG(args...)\
8de5      { std::cerr << "line␣" << __LINE__ << ":␣" << __func__ << "():␣";\
c542      __log(#args, ##args); }
a8cb  #else
```

```
#define LOG(...) //log(a, b, c, ..., xxx);                        459a
#endif                                                            1937
}                                                                 95cf
using namespace debugger;                                         7f4f
```

4.2 Fast IO

```
// #define int __int128                                           427e
namespace io {                                                    4324
  constexpr int BUFFER_SIZE = 1 << 16;                            c5a1
  char buffer[BUFFER_SIZE], *head, *tail;                         d422
                                                                  427e
  char get_char() {                                               34a1
    if (head == tail) {                                           bdf6
      int l = (int) fread(buffer, 1, BUFFER_SIZE, stdin);         c9cd
      tail = (head = buffer) + l;                                 ae81
    }                                                             95cf
    return *head++;                                               02cc
  }                                                               95cf
                                                                  427e
  int read() {                                                    05b1
    int x = 0, f = 1;                                             2544
    char c = get_char();                                          edf0
    for (; !isdigit(c); c = get_char())                           7a2d
      if (c == '-') f = -1;                                       95cc
    for (; isdigit(c); c = get_char()) x = x * 10 + c - '0';      f21e
    return x * f;                                                 827a
  }                                                               95cf
                                                                  427e
  void print(int x) {                                             e794
    if (x > 9) print(x / 10);                                     44c3
    putchar(x % 10 | '0');                                        b830
  }                                                               95cf
                                                                  427e
  void println(int x) { print(x), putchar('\n'); }                5037
}  // namespace io                                                95cf
using namespace io;                                               b96b
```