```cpp
const int MAXN = 1010;
const int SQRTN = 35;
int v[MAXN];
int blocks[SQRTN + 10];
int n;

void build sqrt() {
    SET(blocks);
    FOR0(i,n) {
        int b = i/SQRTN;
        if(blocks[b]==-1 || v[i]<v[blocks[b]]) {
            blocks[b] = i;
        }
    }
}

void update sqrt(int x, int vx) {
    v[x] = vx;
    int b = x/SQRTN;
    if(vx < v[blocks[b]]) blocks[b] = x;
}

int query sqrt(int x, int y) {
    int ans;
    int minv = INF;
    int i = x;
    while(i<=y && i%SQRTN) {
        if(v[i]<minv) {
            minv = v[i];
            ans = i;
        }
        ++i;
    }
    while(i+SQRTN-1 <= y) {
        int b = i/SQRTN;
        if(v[blocks[b]] < minv) {
            minv = v[blocks[b]];
            ans = blocks[b];
        }
        i += SQRTN;
    }
    while(i<=y) {
        if(v[i]<minv) {
            minv = v[i];
            ans = i;
        }
        ++i;
    }
    return ans;
}

int main() {
    NSYNC;
    n = 1000;
    set<int> st;
    srand(time(NULL));
    FOR0(i,n) {
        int x;
        while(true) {
            x = rand()%1000000;
            if(!st.count(x)) break;
        }
        st.insert(x);
        v[i] = x;
    }
    build sqrt();
    int q = 100;
    while(q--) {
        int x = rand()%n;
        int y = rand()%n;
```

```
71          if(x>y) swap(x,y);
72          int minv = INF;
73          int ans;
74          FOR(i,x,y+1) {
75              if(v[i] < minv) {
76                  minv = v[i];
77                  ans = i;
78              }
79          }
80          if(ans!=query sqrt(x,y)) {
81              DBGP(mp(x,y));
82              DBG(ans);
83              DBG(query sqrt(x,y));
84              cout << "WA\n";
85              exit(0);
86          }
87      }
88      cout << "Accepted\n";
89      return 0;
90  }
91
```