```cpp
struct node
{
    node *left, *right;
    node()
    {
        left = right = NULL;
    }
};

inline node merge(node &L, node &R)
{
    node ans;
    // here
    return ans;
}

node query(node *root, int i, int j, int L, int R)
{
    if(L <= i and R >= j) return *root;
    if(j < L || i > R) return node();
    int mid = (i+j)/2;
    node left =  query(root->left, i, mid, L, R);
    node right = query(root->right, mid+1, j, L, R);
    return merge(left,right);
}

void update(node *root, int i, int j, int pos, int num)
{
    if(i == j)
    {
        // here
        return;
    }
    int mid = (i+j)/2;
    if(pos <= mid) update(root->left,i,mid,pos,num);
    else update(root->right,mid+1,j,pos,num);
    node res = merge(*(root->left),*(root->right));
    // copy res to *root
}

void build(node *root, int i, int j)
{
    if(i == j)
    {
        // here
        return;
    }
    root->left = new node;
    root->right = new node;
    int mid = (i+j)/2;
    build(root->left,i,mid);
    build(root->right,mid+1,j);
```

```
    node m = merge(*(root->left),*(root->right));
    // copy m to *root
}
```