Min Max Recursive:

#include <bits/stdc++.h>

using namespace std;

void minmax(vector<int> &v, int l, int h, int \*mn, int \*mx)

{

if(l == h)

{

\*mn = v[l];

\*mx = v[h];

}

else if(l + 1 == h)

{

if(v[l] <= v[h])

{

\*mn = v[l];

\*mx = v[h];

}

else

{

\*mn = v[h];

\*mx = v[l];

}

}

else

{

int mid = (l + h)/2;

int mn1 = 1e9, mn2 = 1e9, mx1 = -1e9, mx2 = 1e9;

minmax(v, l, mid, &mn1, &mx1);

minmax(v, mid + 1, h, &mn2, &mx2);

if(mn1 < mn2)

{

\*mn = mn1;

}

else

{

\*mn = mn2;

}

if(mx1 > mx2)

{

\*mx = mx1;

}

else

{

\*mx = mx2;

}

}

}

int main() {

int n;

cout << "Enter number of elements\n";

cin >> n;

vector<int> v(n);

cout << "Enter the elements\n";

for(auto &i: v)cin >> i;

int mn =1e9, mx = -1;

minmax(v, 0, v.size() - 1, &mn, &mx);

cout << "Min: " << mn << " Max: " <<mx << "\n";

return 0;

}

**ITERATIVE**

#include <bits/stdc++.h>

using namespace std;

void minmax(vector<int> &v, int l, int h, int \*mn, int \*mx)

{

if(l == h)

{

\*mn = v[l];

\*mx = v[h];

}

else if(l + 1 == h)

{

if(v[l] <= v[h])

{

\*mn = v[l];

\*mx = v[h];

}

else

{

\*mn = v[h];

\*mx = v[l];

}

}

else

{

int mid = (l + h)/2;

int mn1 = 1e9, mn2 = 1e9, mx1 = -1e9, mx2 = 1e9;

minmax(v, l, mid, &mn1, &mx1);

minmax(v, mid + 1, h, &mn2, &mx2);

if(mn1 < mn2)

{

\*mn = mn1;

}

else

{

\*mn = mn2;

}

if(mx1 > mx2)

{

\*mx = mx1;

}

else

{

\*mx = mx2;

}

}

}

void minmax\_iter(vector<int> &v, int \*mn, int \*mx)

{

int mna = 1e9;

int mxa = -1e9;

for(auto &i: v)

{

mna = min(mna, i);

mxa = max(mxa, i);

}

\*mn = mna;

\*mx = mxa;

}

int main() {

int n;

cout << "Enter number of elements\n";

cin >> n;

vector<int> v(n);

cout << "Enter the elements\n";

for(auto &i: v)cin >> i;

int mn =1e9, mx = -1;

minmax(v, 0, v.size() - 1, &mn, &mx);

minmax\_iter(v, &mn, &mx);

cout << "Min: " << mn << " Max: " <<mx << "\n";

return 0;

}