Find the largest rectangular area possible in a given histogram where the largest rectangle can be made of a number of contiguous bars.

Input Format

First line contains a single integer N, denoting the number of bars in th histogram. Next line contains N integers, ith of which, denotes the height of ith bar in the histogram.

Constraints

1<=N<=10^6 Height of each bar in histogram <= 10^9

Output Format

Output a single integer denoting the area of the required rectangle.

Sample Input

5

1 2 3 4 5

Sample Output

9

Program-

#include<iostream>

#include<stack>

using namespace std;

#define ll long long int

ll getMaxArea(ll hist[], ll n)

{

stack<ll> s;

ll max\_area = 0;

ll tp;

ll area\_with\_top;

ll i = 0;

while (i < n)

{

if (s.empty() || hist[s.top()] <= hist[i])

s.push(i++);

else

{

while (!s.empty() && hist[s.top()] > hist[i]) {

tp = s.top();

s.pop();

area\_with\_top = hist[tp] \* (s.empty() ? i : i - s.top() - 1);

if (max\_area < area\_with\_top)

max\_area = area\_with\_top;

}

s.push(i++);

}

}

while (!s.empty()) {

tp = s.top();

s.pop();

area\_with\_top = hist[tp] \* (s.empty() ? i : i - s.top() - 1);

if (max\_area < area\_with\_top)

max\_area = area\_with\_top;

}

return max\_area;

}

int main()

{

ll n,i;

cin>>n;

ll hist[n];

for(i=0;i<n;i++){

cin>>hist[i];

}

cout << getMaxArea(hist, n);

return 0;

}