The stock span problem is a financial problem where we have a series of N daily price quotes for a stock and we need to calculate span of stock’s price for all N days. You are given an array of length N, where ith element of array denotes the price of a stock on ith. Find the span of stock's price on ith day, for every 1<=i<=N.

A span of a stock's price on a given day, i, is the maximum number of consecutive days before the (i+1)th day, for which stock's price on these days is less than or equal to that on the ith day.

Input Format

Enter the size of the array N and add N more numbers and store in the array.

Constraints

None

Output Format

Display the array containing stock span values.

Sample Input

5

30

35

40

38

35

Sample Output

1 2 3 1 1 END

For the given case

for day1 stock span =1 for day2 stock span =2 (as 35>30 so both days are included in it) for day3 stock span =3 (as 40>35 so 2+1=3) for day4 stock span =1 (as 38<40 so only that day is included) for day5 stock span =1 (as 35<38 so only that day is included) hence output is 1 2 3 1 1 END

Program-

#include <iostream>

#include<stack>

using namespace std;

void calculatespan(int price[],int n,int span[]){

stack<int> s;

s.push(0);

span[0]=1;

for(int i=1;i<n;i++){

while(!s.empty() && price[s.top()]<=price[i])

s.pop();

span[i]=(s.empty()) ? i+1 : (i-s.top());

s.push(i);

}

}

void print(int span[],int n){

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

cout<<span[i]<<" ";

}

cout<<"END";

}

int main()

{

int n,i;

cin>>n;

int price[n],span[n];

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

cin>>price[i];

}

calculatespan(price,n,span);

print(span,n);

}