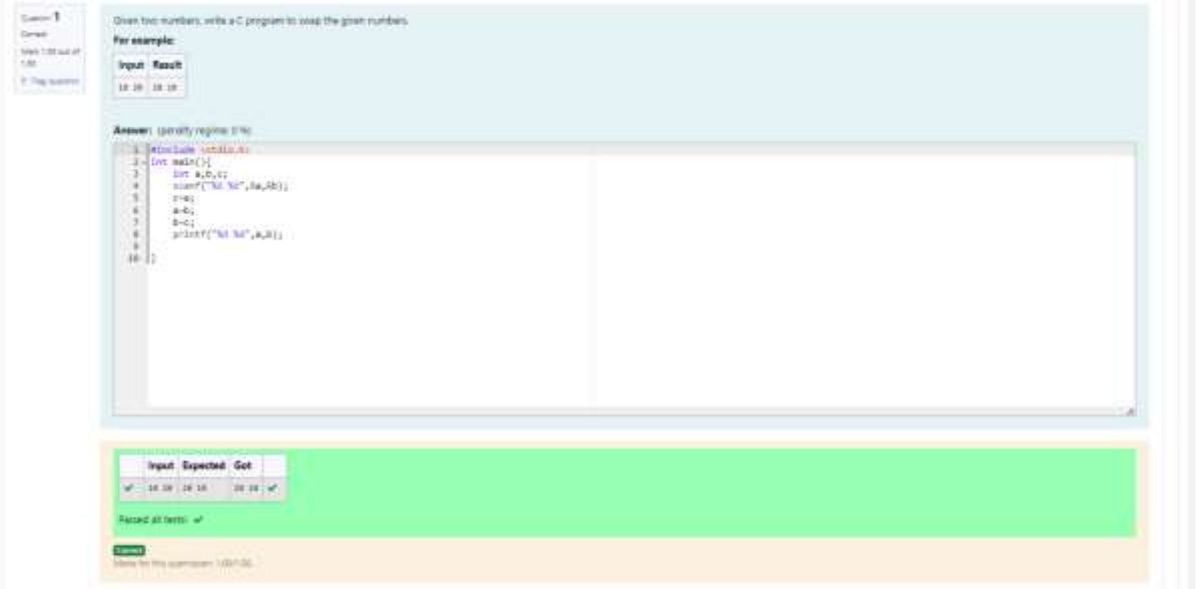
WEEK_1



Given two numbers, write a C program to swap the given numbers.

For example:

Input	Result
10 20	20 10

Answer: (penalty regime: 0 %)

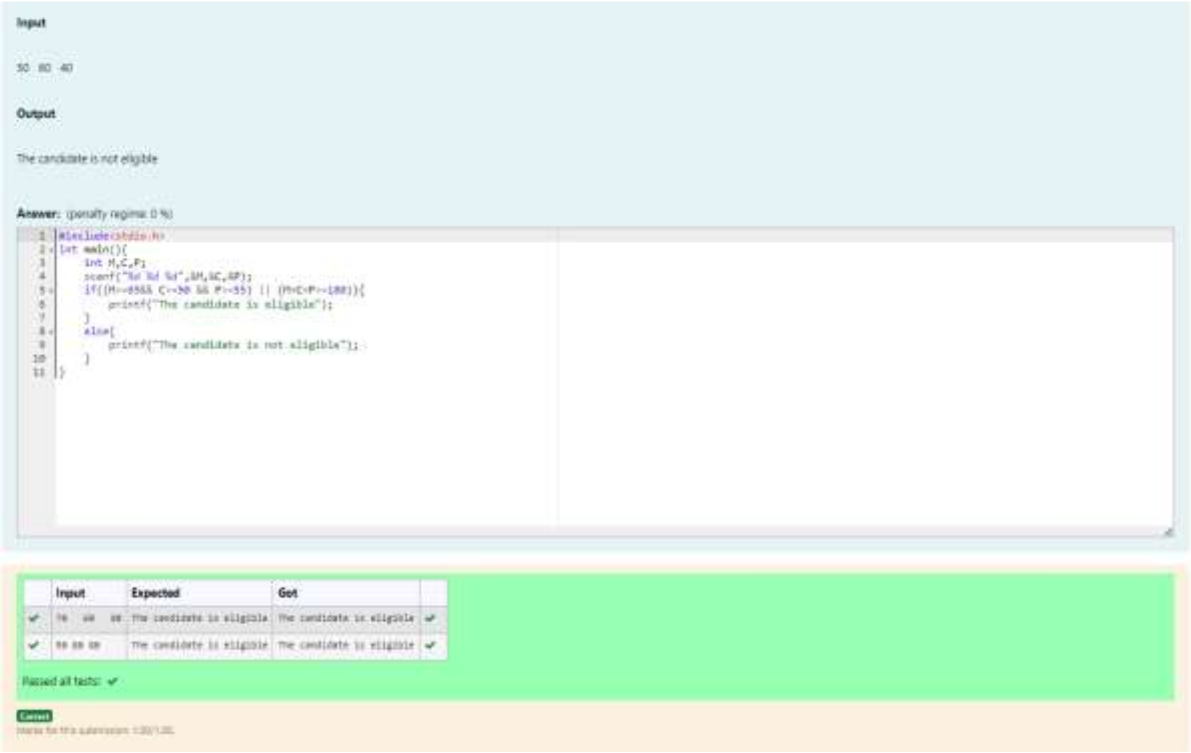
```
#include <stdio.h>
int main(){
    int a,b,c;
    scanf("%d %d",&a,&b);
    c=a;
    a=b;
    b=c;
    printf("%d %d",a,b);

}
```

	Input	Expected	Got	
✓	10 20	20 10	20 10	✓

Passed all tests: ✓

Marks for this submission: 1.00/1.00.

2.



Input

50 80 40

Output

The candidate is not eligible

Answer: (penalty regime: 0 %)

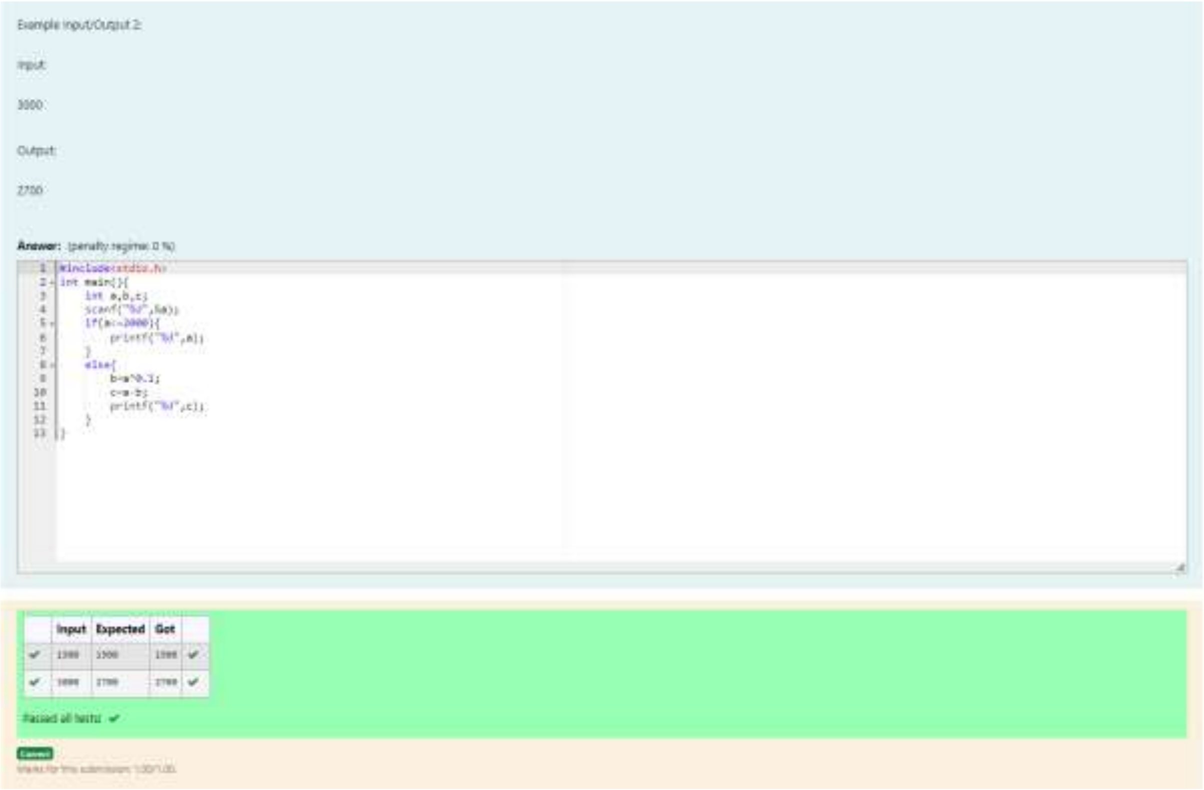
```
#include<stdio.h>
int main(){
    int M,C,P;
    scanf("%d %d %d",&M,&C,&P);
    if((M>=65&& C>=50 && P>=55) || (M+C+P>=180)){
        printf("The candidate is eligible");
    }
    else{
        printf("The candidate is not eligible");
    }
}
```

	Input	Expected	Got	
✓	70 60 80	The candidate is eligible	The candidate is eligible	✓
✓	80 80 80	The candidate is eligible	The candidate is eligible	✓

Passed all tests: ✓

Marks for this submission: 1.00/1.00.

3.



Example Input/Output 2:

Input:

3000

Output:

2700

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
    int a,b,c;
    scanf("%d",&a);
    if(a<=2000){
        printf("%d",a);
    }
    else{
        b=a*0.1;
        c=a-b;
        printf("%d",c);
    }
}
```

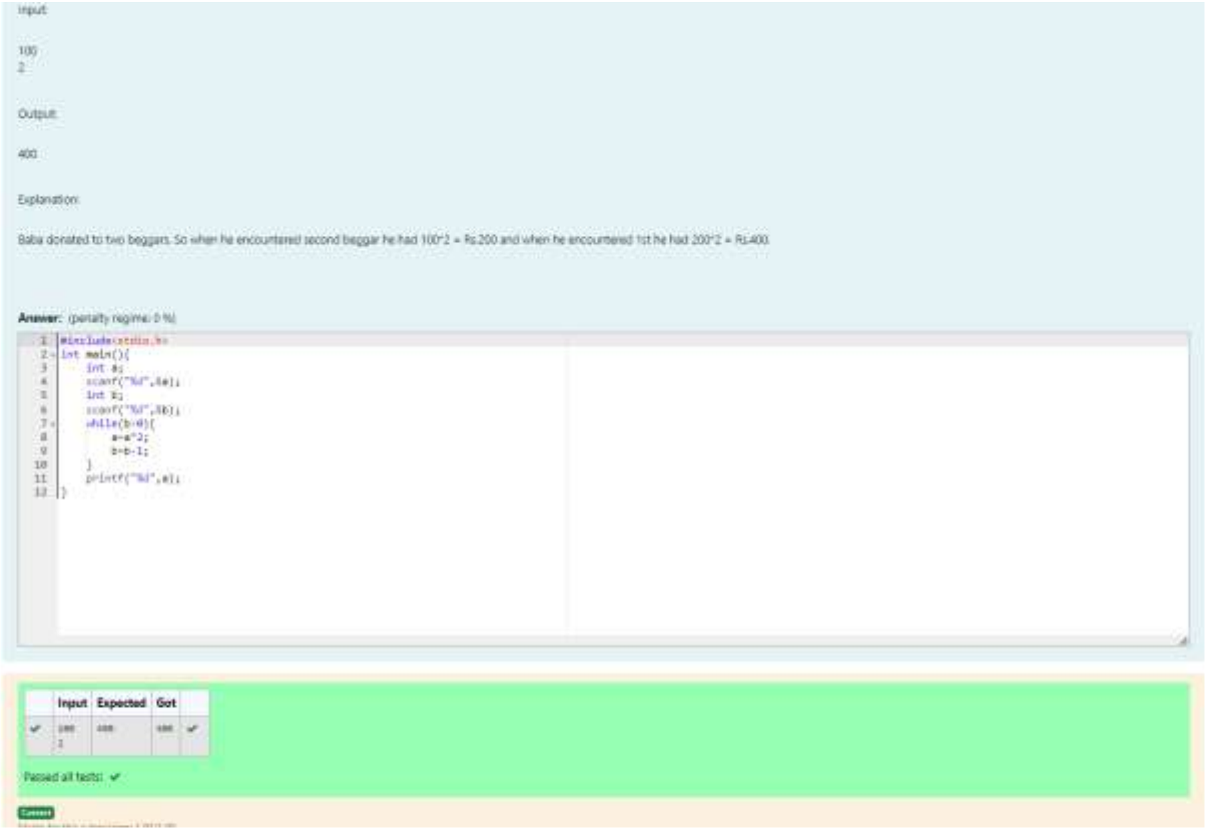
	Input	Expected	Got	
✓	1500	1500	1500	✓
✓	3000	2700	2700	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

4.



Input:

100
2

Output:

400

Explanation:

Baba donated to two beggars. So when he encountered second beggar he had 100*2 = Rs.200 and when he encountered 1st he had 200*2 = Rs.400.

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
    int a;
    scanf("%d",&a);
    int b;
    scanf("%d",&b);
    while(b>0){
        a=a*2;
        b=b-1;
    }
    printf("%d",a);
}
```

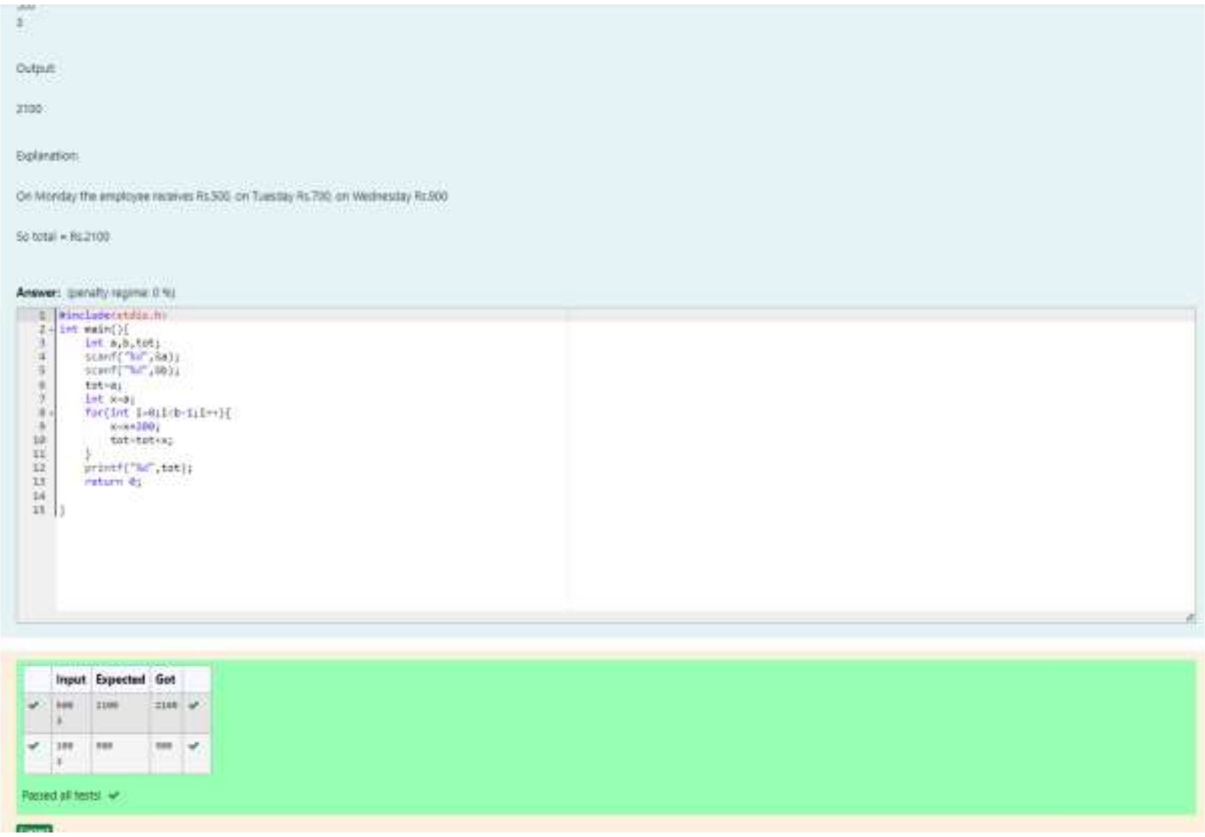
	Input	Expected	Got	
✓	100 2	400	400	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

5.



500
3

Output:

2100

Explanation:

On Monday the employee receives Rs.500 on Tuesday Rs.700 on Wednesday Rs.900

So total = Rs.2100

Answer: (penalty regime: 0 %)

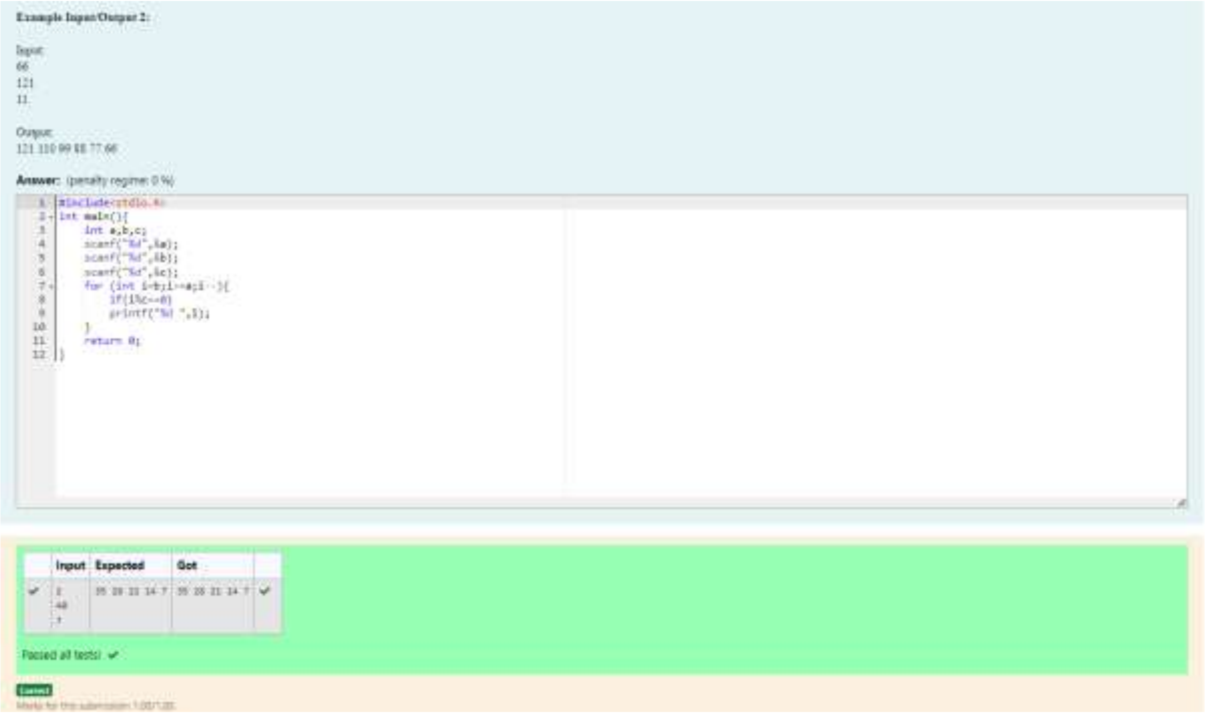
```c
#include<stdio.h>
int main(){
    int a,b,tot;
    scanf("%d",&a);
    scanf("%d",&b);
    tot=a;
    int x=a;
    for(int i=0;i<b-1;i++){
        x=x+200;
        tot=tot+x;
    }
    printf("%d",tot);
    return 0;

}
```

	Input	Expected	Got	
✓	500 3	2100	2100	✓
✓	100 3	900	900	✓

Passed all tests! ✓

Correct

6.



Example Input/Output 2:

Input:
66
121
11

Output:
121 110 99 88 77 66

Answer: (penalty regime: 0 %)

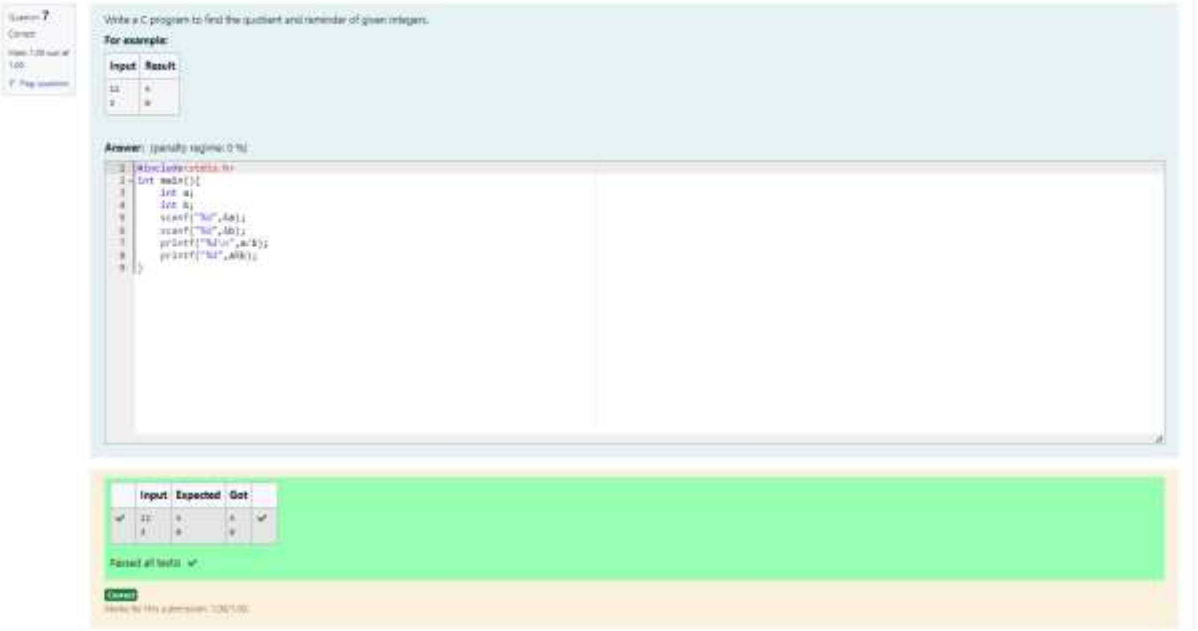
```c
#include<stdio.h>
int main(){
    int a,b,c;
    scanf("%d",&a);
    scanf("%d",&b);
    scanf("%d",&c);
    for (int i=b;i>=a;i--){
        if(i%c==0)
        printf("%d ",i);
    }
    return 0;
}
```

	Input	Expected	Got	
✓	2 40 7	35 28 21 14 7	35 28 21 14 7	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 7

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the quotient and reminder of given integers.

For example:

Input	Result
12 3	4 0

Answer: (penalty regime: 0 %)

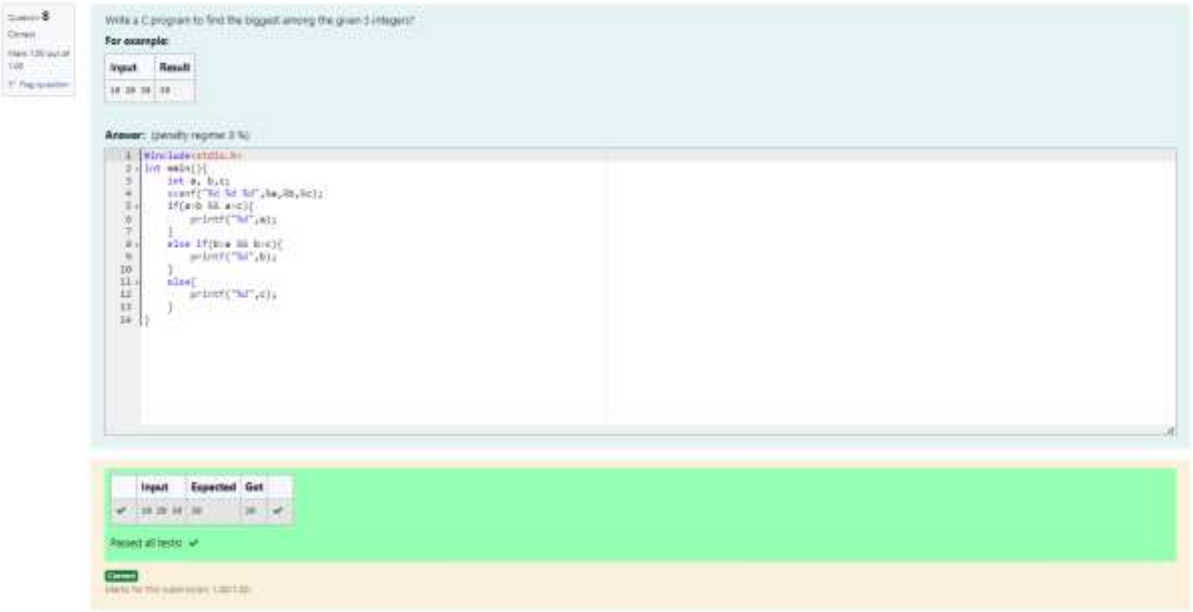
```c
#include<stdio.h>
int main(){
    int a;
    int b;
    scanf("%d",&a);
    scanf("%d",&b);
    printf("%d\n",a/b);
    printf("%d",a%b);
}
```

	Input	Expected	Got	
✓	12 3	4 0	4 0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 8

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the biggest among the given 3 integers?

For example:

Input	Result
10 20 30	30

Answer: (penalty regime: 0 %)

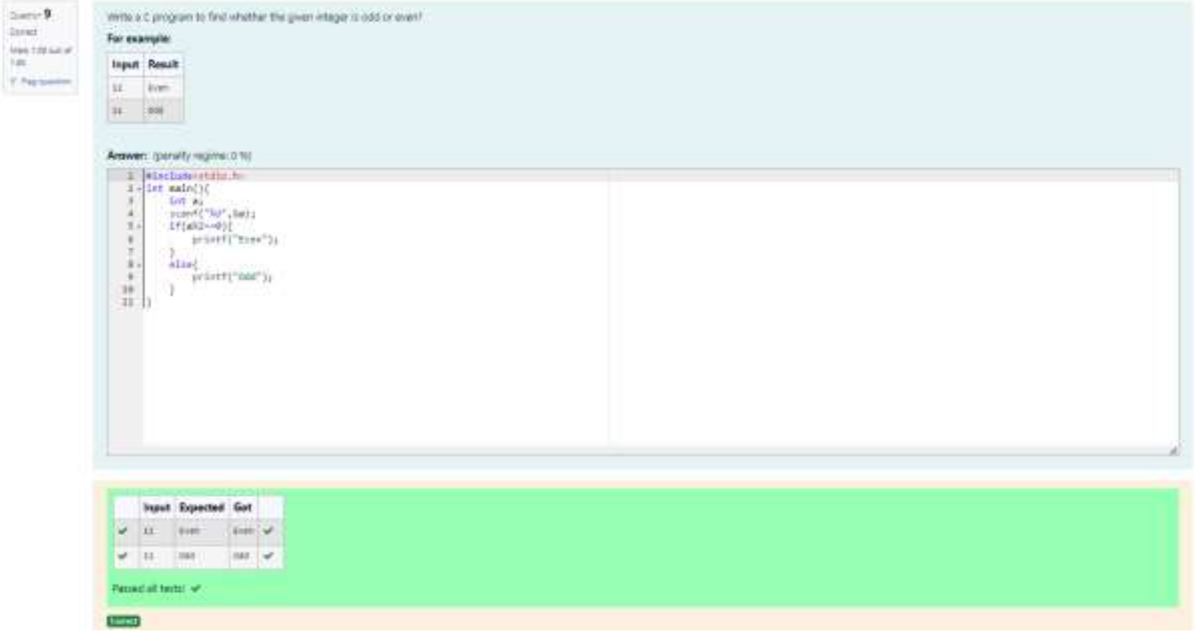
```c
#include<stdio.h>
int main(){
    int a, b,c;
    scanf("%d %d %d",&a,&b,&c);
    if(a>b && a>c){
        printf("%d",a);
    }
    else if(b>a && b>c){
        printf("%d",b);
    }
    else{
        printf("%d",c);
    }
}
```

	Input	Expected	Got	
✓	10 20 30	30	30	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 9

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find whether the given integer is odd or even?

For example:

Input	Result
12	Even
11	Odd

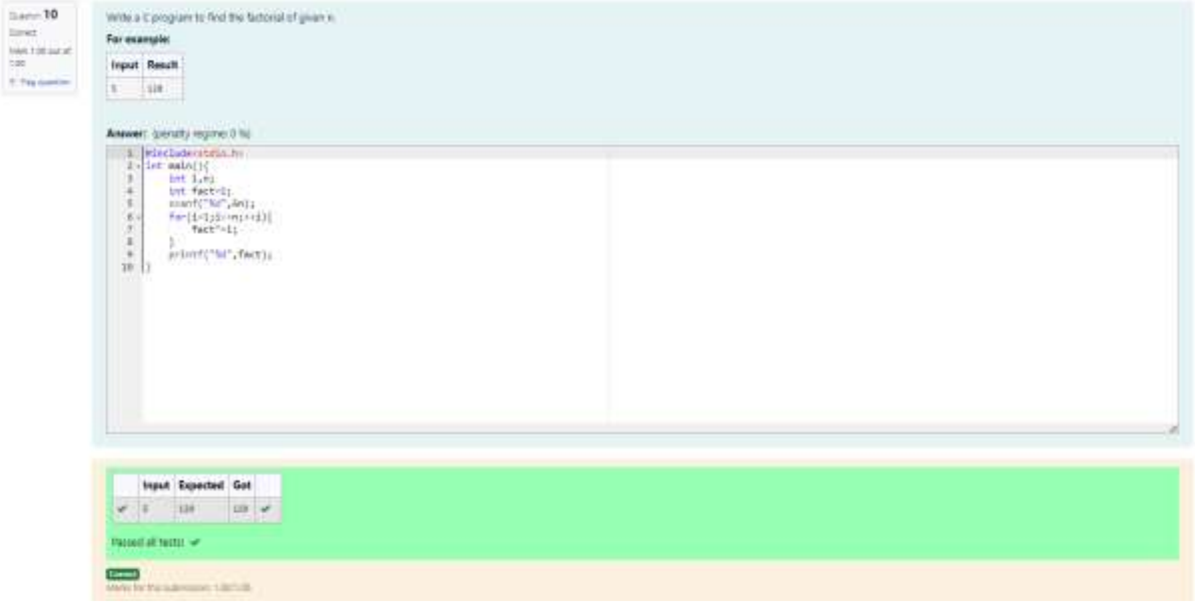
Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
    int a;
    scanf("%d",&a);
    if(a%2==0){
        printf("Even");
    }
    else{
        printf("Odd");
    }
}
```

	Input	Expected	Got	
✓	12	Even	Even	✓
✓	11	Odd	Odd	✓

Passed all tests! ✓

Correct



Question 10

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the factorial of given n.

For example:

Input	Result
5	120

Answer: (penalty regime: 0 %)

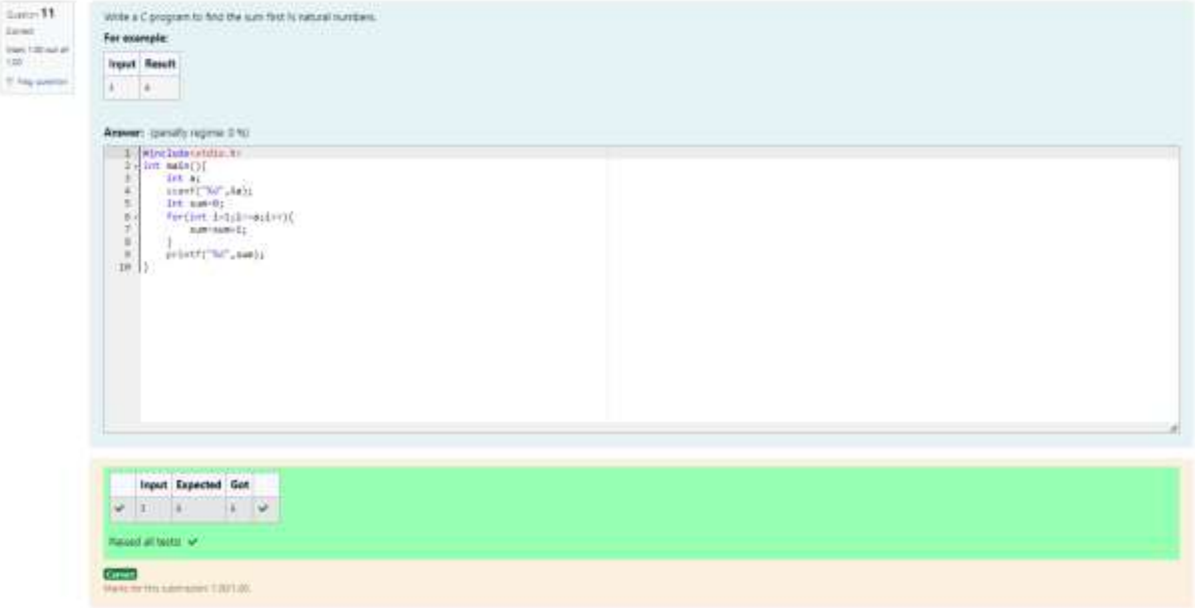
```
#include<stdio.h>
int main(){
    int i,n;
    int fact=1;
    scanf("%d",&n);
    for(i=1;i<=n;++i){
        fact*=i;
    }
    printf("%d",fact);
}
```

	Input	Expected	Got	
✓	5	120	120	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 11

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the sum first N natural numbers.

For example:

Input	Result
3	6

Answer: (penalty regime: 0 %)

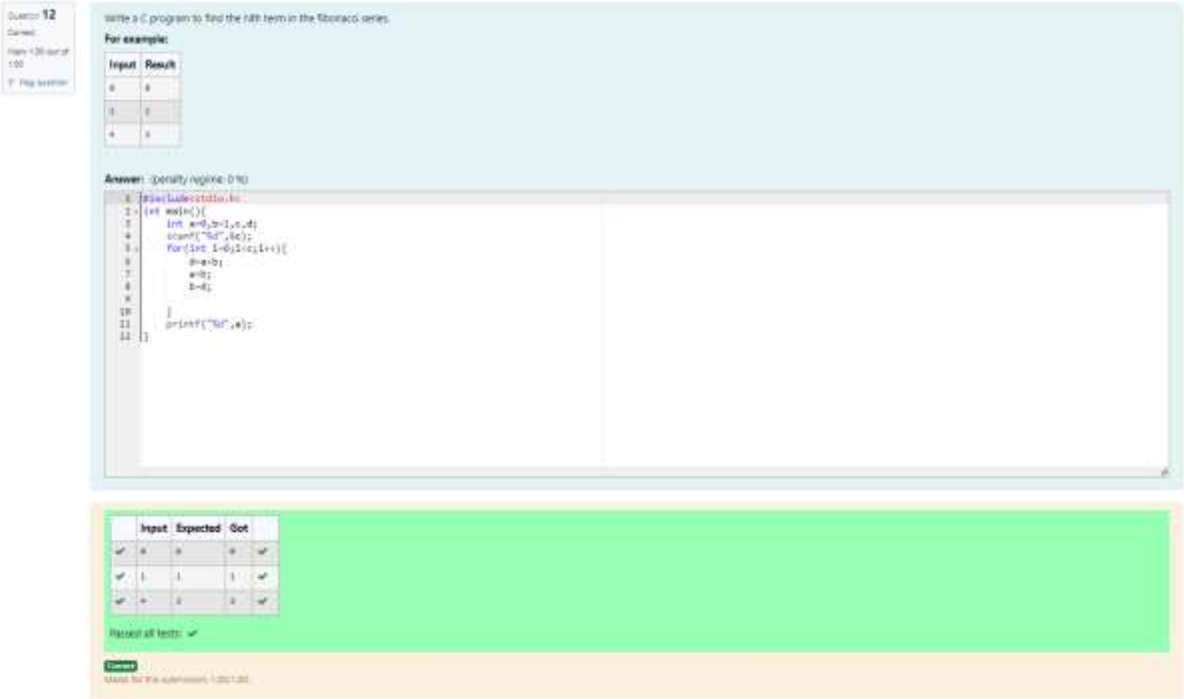
```
#include<stdio.h>
int main(){
    int a;
    scanf("%d",&a);
    int sum=0;
    for(int i=1;i<=a;i++){
        sum=sum+i;
    }
    printf("%d",sum);
}
```

	Input	Expected	Got	
✓	3	6	6	✓

Passed all tests: ✓

Correct

Marks for this submission: 1.00/1.00.



Question 12

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the nth term in the fibonacci series.

For example:

Input	Result
0	0
1	1
4	3

Answer: (penalty regime: 0 %)

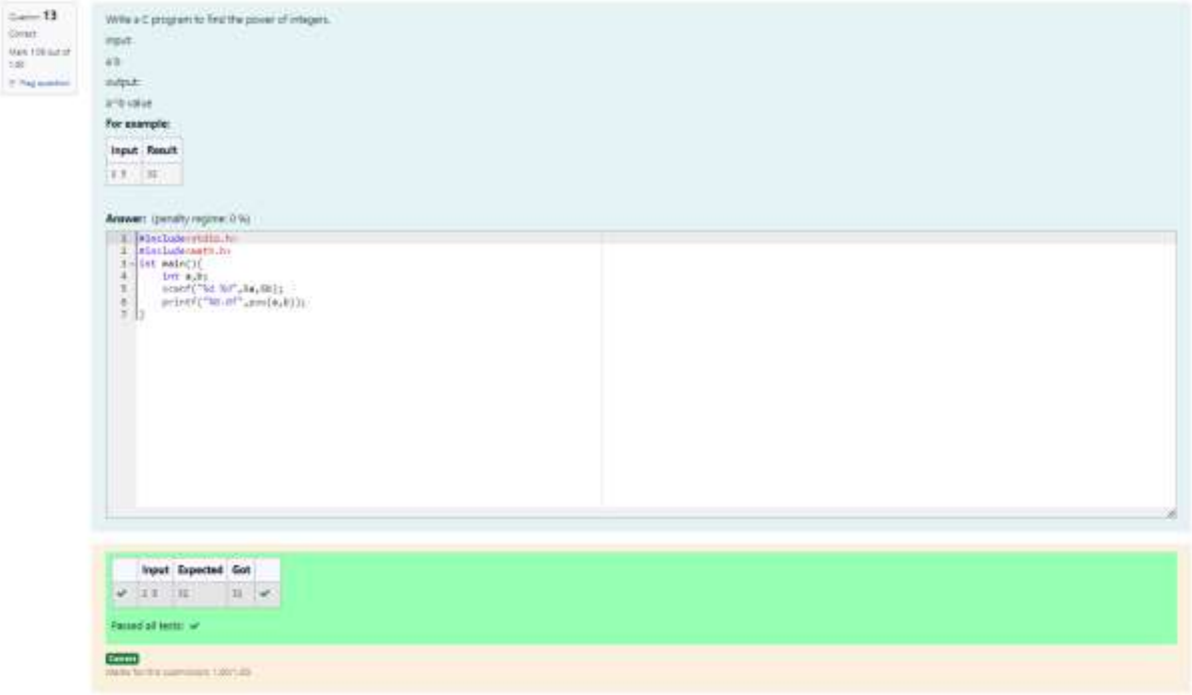
```
#include<stdio.h>
int main(){
    int a=0,b=1,c,d;
    scanf("%d",&c);
    for(int i=0;i<c;i++){
        d=a+b;
        a=b;
        b=d;

    }
    printf("%d",a);
}
```

	Input	Expected	Got	
✓	0	0	0	✓
✓	1	1	1	✓
✓	4	3	3	✓

Passed all tests: ✓

Correct

Marks for this submission: 1.00/1.00.



Question **13**

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find the power of integers.

input:

a b

output:

a^b value

For example:

Input	Result
2 5	32

Answer: (penalty regime: 0 %)

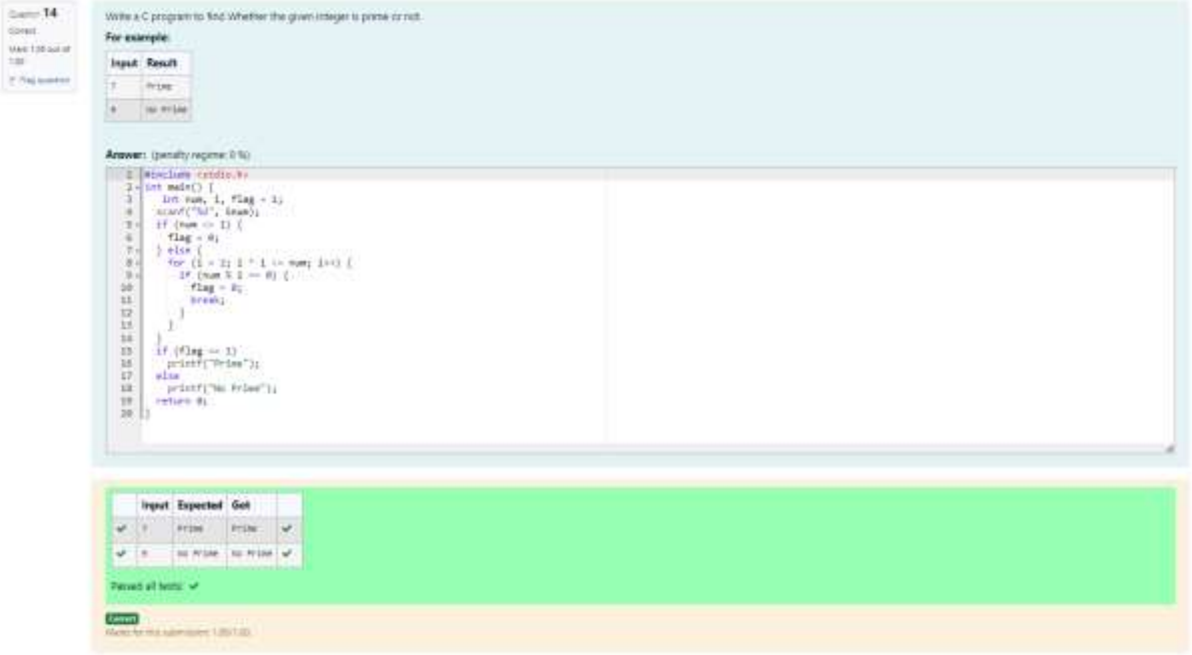
```c
#include<stdio.h>
#include<math.h>
int main(){
    int a,b;
    scanf("%d %d",&a,&b);
    printf("%.0f",pow(a,b));
}
```

	Input	Expected	Got	
✓	2 5	32	32	✓

Passed all tests: ✓

Correct

Marks for this submission: 1.00/1.00.



Question **14**

Correct

Mark 1.00 out of 1.00

Flag question

Write a C program to find Whether the given integer is prime or not.

For example:

Input	Result
7	Prime
8	No Prime

Answer: (penalty regime: 0 %)

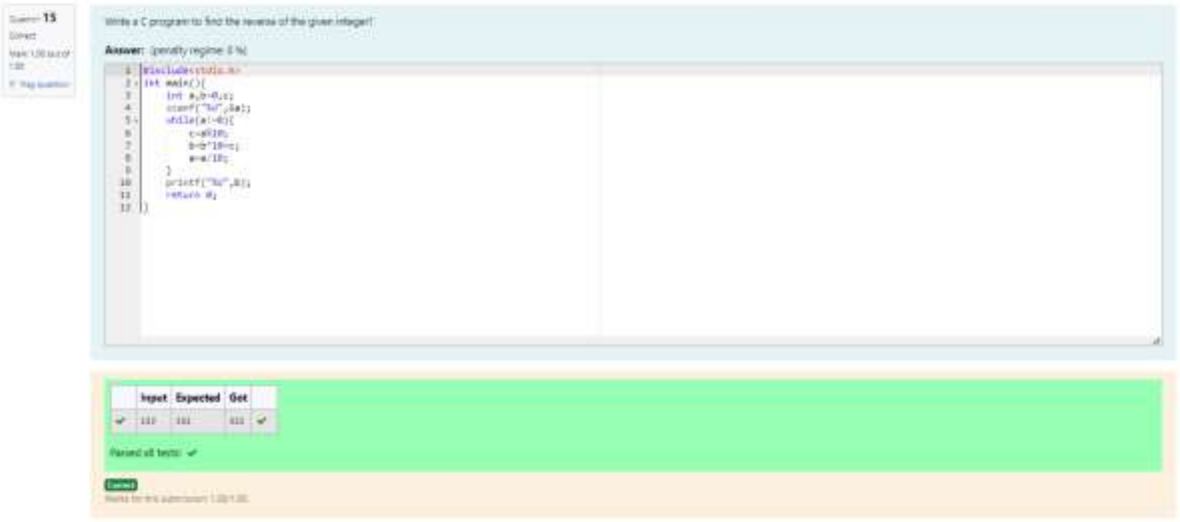
```c
#include <stdio.h>
int main() {
   int num, i, flag = 1;
  scanf("%d", &num);
  if (num <= 1) {
    flag = 0;
  } else {
    for (i = 2; i * i <= num; i++) {
      if (num % i == 0) {
        flag = 0;
        break;
      }
    }
  }
  if (flag == 1)
    printf("Prime");
  else
    printf("No Prime");
  return 0;
}
```

	Input	Expected	Got	
✓	7	Prime	Prime	✓
✓	8	No Prime	No Prime	✓

Passed all tests: ✓

Correct

Marks for this submission: 1.00/1.00.



Write a C program to find the reverse of the given integer?

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main(){
    int a,b=0,c;
    scanf("%d",&a);
    while(a!=0){
        c=a%10;
        b=b*10+c;
        a=a/10;
    }
    printf("%d",b);
    return 0;
}
```

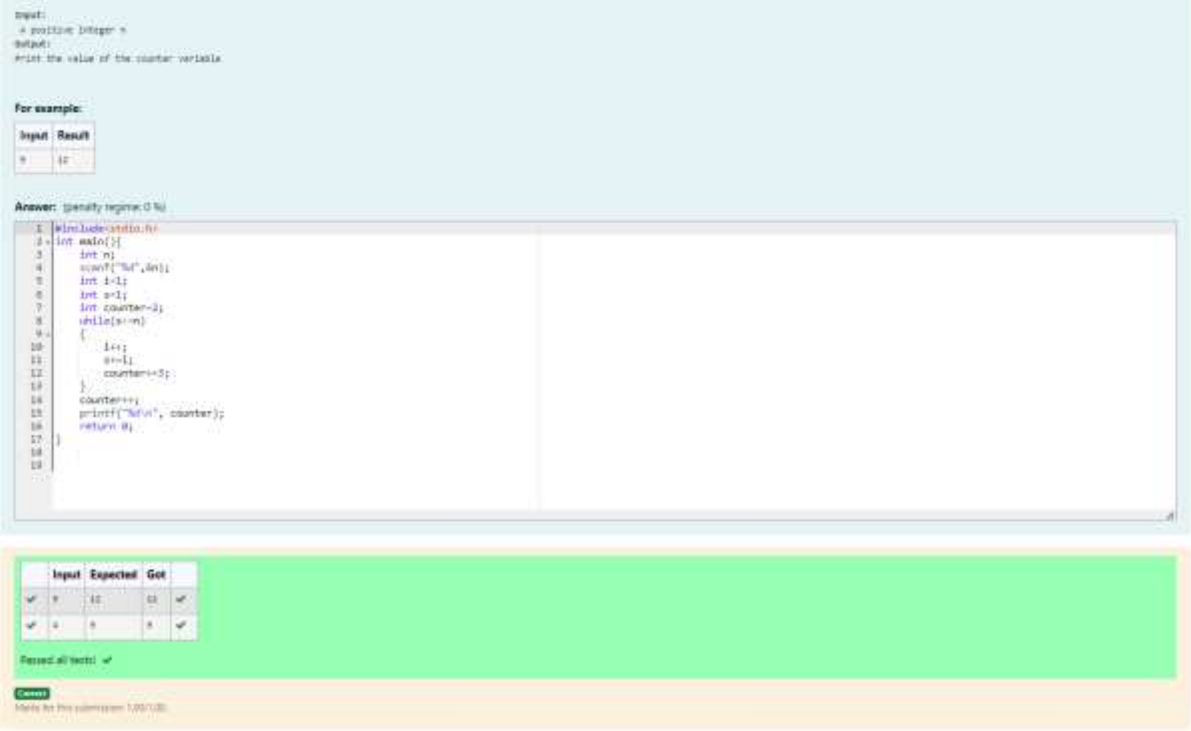
	Input	Expected	Got	
✓	123	321	321	✓

Passed all tests: ✓

Marks for this submission: 1.00/1.00.

TIME COMPLEXITY

1.



```
Input:
 A positive integer n
Output:
Print the value of the counter variable
```

For example:

Input	Result
9	12

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main(){
    int n;
    scanf("%d",&n);
    int i=1;
    int s=1;
    int counter=2;
    while(s<=n)
    {
        i++;
        s+=i;
        counter+=3;
    }
    counter++;
    printf("%d\n", counter);
    return 0;
}
```

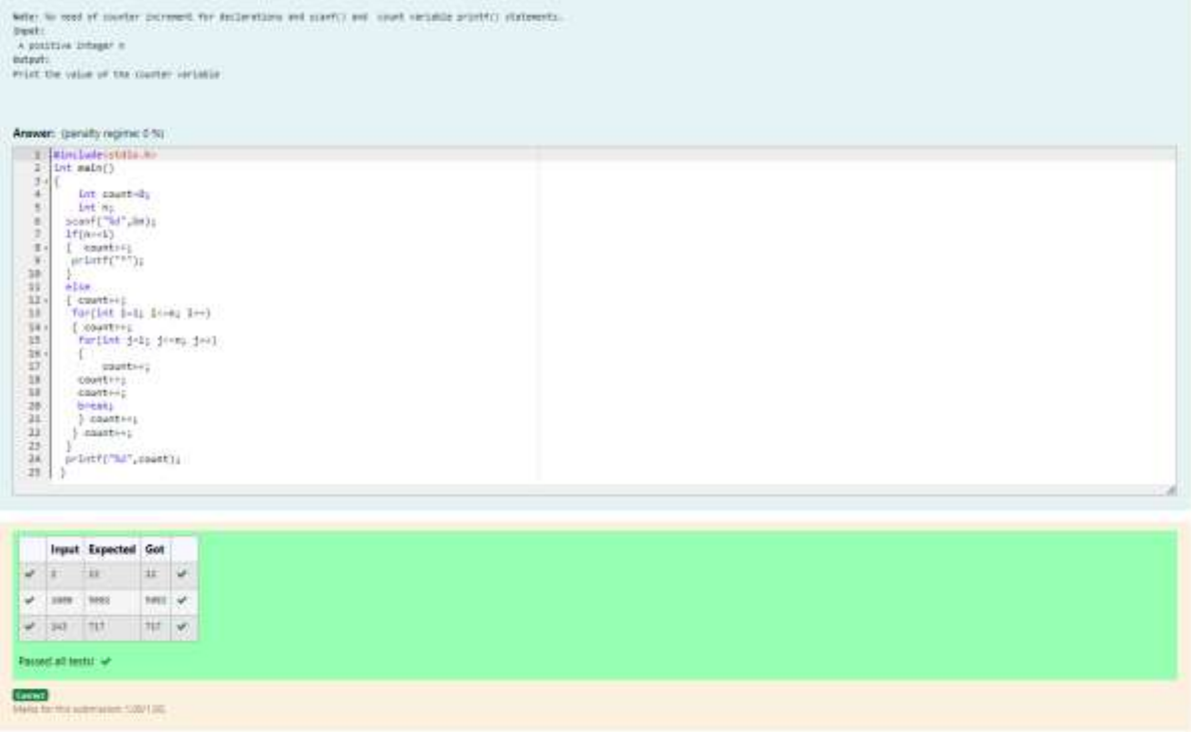
	Input	Expected	Got	
✓	9	12	12	✓
✓	4	9	9	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

2.



```
Note: No need of counter increment for declarations and scanf() and  count variable printf() statements.
Input:
 A positive integer n
Output:
Print the value of the counter variable
```

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main()
{
    int count=0;
    int n;
  scanf("%d",&n);
  if(n==1)
  {  count++;
   printf("*");
  }
  else
  { count++;
   for(int i=1; i<=n; i++)
   { count++;
    for(int j=1; j<=n; j++)
    {
        count++;
    count++;
    count++;
    break;
    } count++;
   } count++;
  }
  printf("%d",count);
 }
```

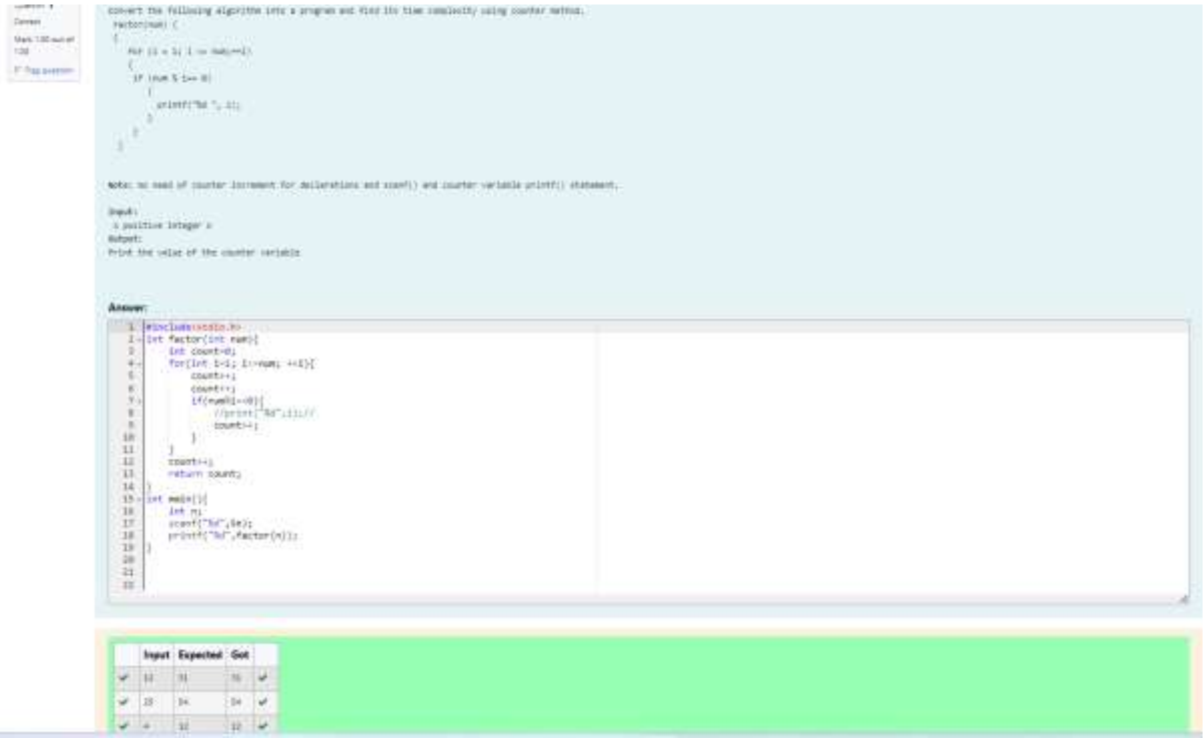
	Input	Expected	Got	
✓	2	12	12	✓
✓	1000	5002	5002	✓
✓	143	717	717	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

3.



Convert the following algorithm into a program and find its time complexity using counter method.

```
 factor(num) {
 {
   for (i = 1; i <= num;++i)
   {
    if (num % i== 0)
      {
       printf("%d ", i);
      }
    }
  }
```

Note: no need of counter increment for declarations and scanf() and counter variable printf() statement.

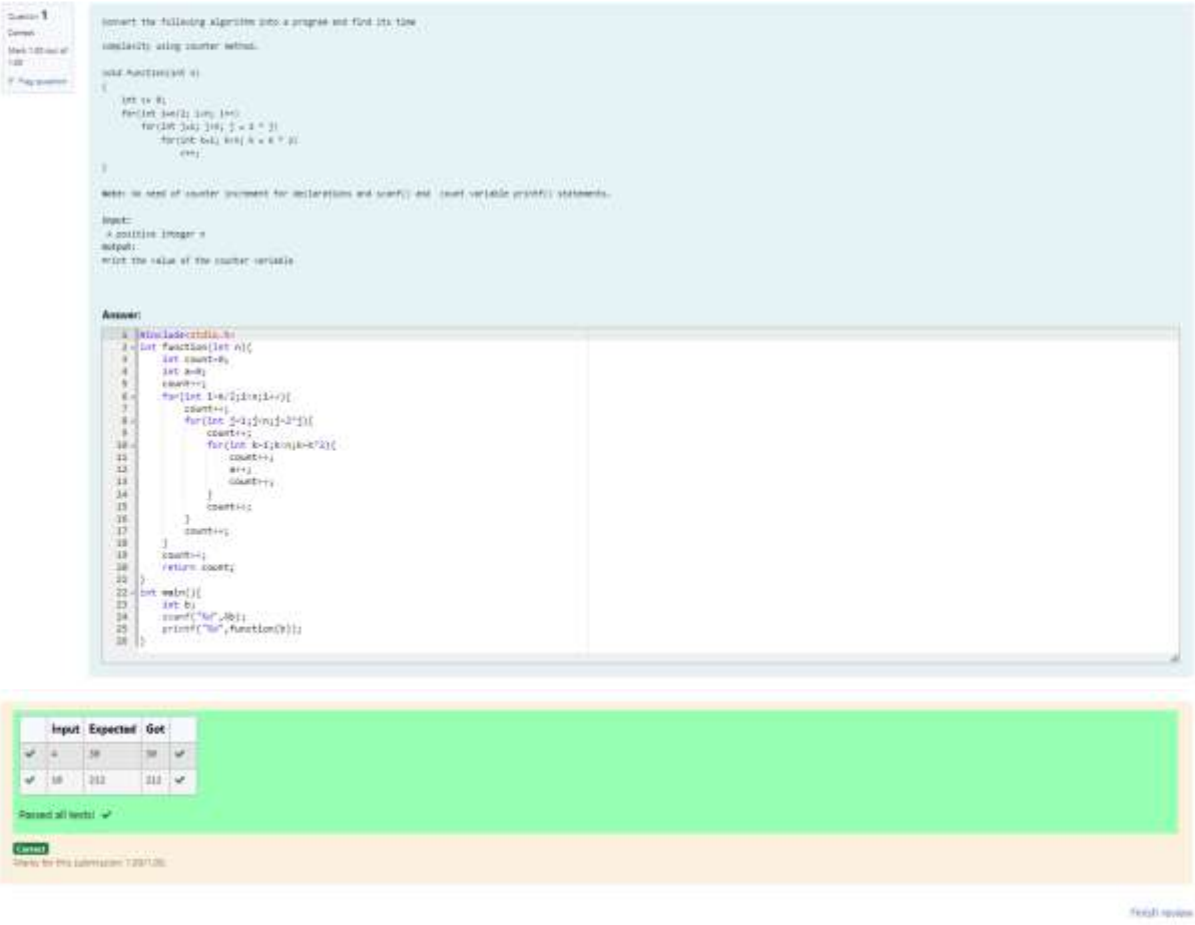
Input:
 A positive integer n
Output:
Print the value of the counter variable

Answer:

```
#include<stdio.h>
int factor(int num){
    int count=0;
    for(int i=1; i<=num; ++i){
        count++;
        count++;
        if(num%i==0){
            //printf("%d",i);//
            count++;
        }
    }
    count++;
    return count;
}
int main(){
    int n;
    scanf("%d",&n);
    printf("%d",factor(n));
}
```

	Input	Expected	Got	
✓	12	31	31	✓
✓	25	54	54	✓
✓	4	12	12	✓

4.



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

```
Convert the following algorithm into a program and find its time

complexity using counter method.

void function(int n)
{
    int i= 0;
    for(int i=n/2; i<n; i++)
        for(int j=1; j<n; j = 2 * j)
            for(int k=1; k<n; k = k * 2)
                c++;
}

Note: No need of counter increment for declarations and scanf() and  count variable printf() statements.

Input:
 A positive Integer n
Output:
Print the value of the counter variable
```

Answer:

```
#include<stdio.h>
int function(int n){
    int count=0;
    int a=0;
    count++;
    for(int i=n/2;i<n;i++){
        count++;
        for(int j=1;j<n;j=2*j){
            count++;
            for(int k=1;k<n;k=k*2){
                count++;
                a++;
                count++;
            }
            count++;
        }
        count++;
    }
    count++;
    return count;
}
int main(){
    int b;
    scanf("%d",&b);
    printf("%d",function(b));
}
```

	Input	Expected	Got	
✓	4	30	30	✓
✓	10	212	212	✓

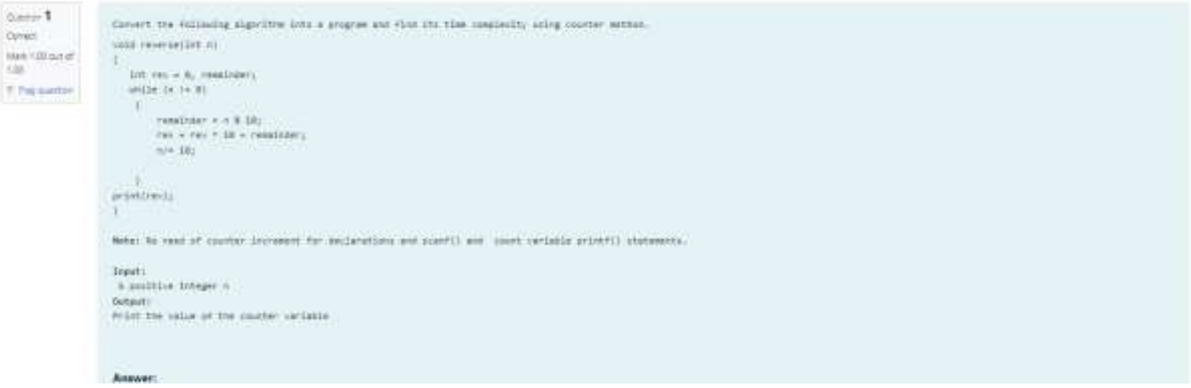
Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Finish review

5.



Question 1

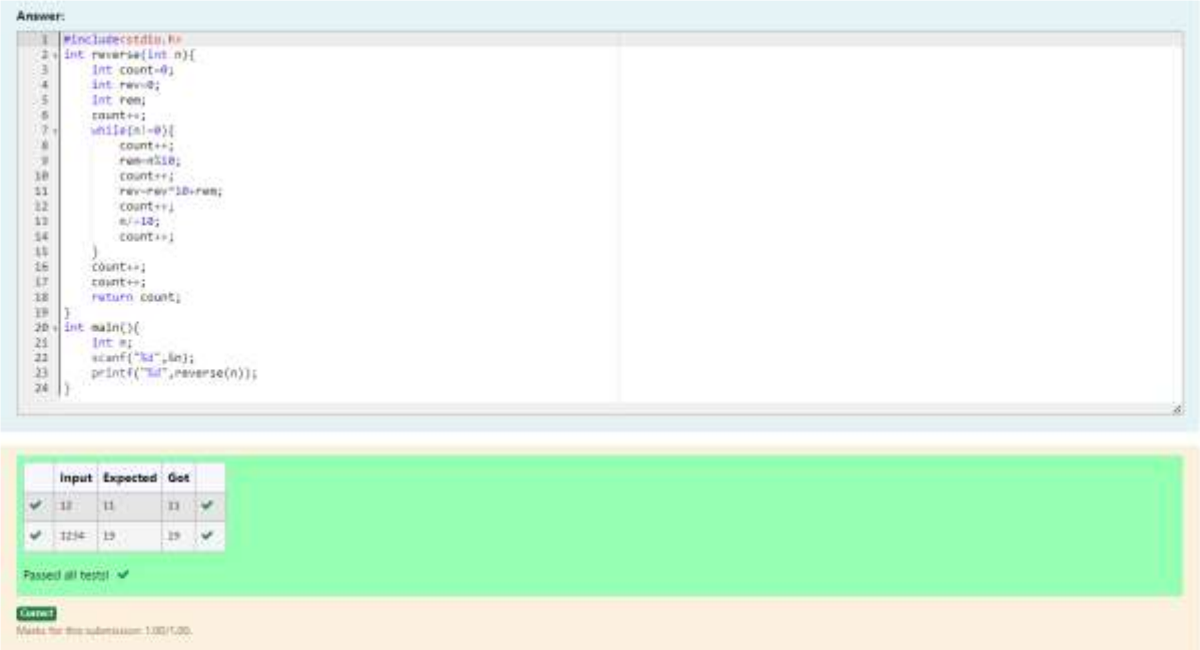
Correct

Mark 1.00 out of 1.00

Flag question

```
Convert the following algorithm into a program and find its time complexity using counter method.
void reverse(int n)
{
   int rev = 0, remainder;
   while (n != 0)
    {
       remainder = n % 10;
       rev = rev * 10 + remainder;
       n/= 10;

    }
print(rev);
}

Note: No need of counter increment for declarations and scanf() and  count variable printf() statements.

Input:
 A positive Integer n
Output:
Print the value of the counter variable
```

Answer:



Answer:

```
#include<stdio.h>
int reverse(int n){
    int count=0;
    int rev=0;
    int rem;
    count++;
    while(n!=0){
        count++;
        rem=n%10;
        count++;
        rev=rev*10+rem;
        count++;
        n/=10;
        count++;
    }
    count++;
    count++;
    return count;
}
int main(){
    int n;
    scanf("%d",&n);
    printf("%d",reverse(n));
}
```

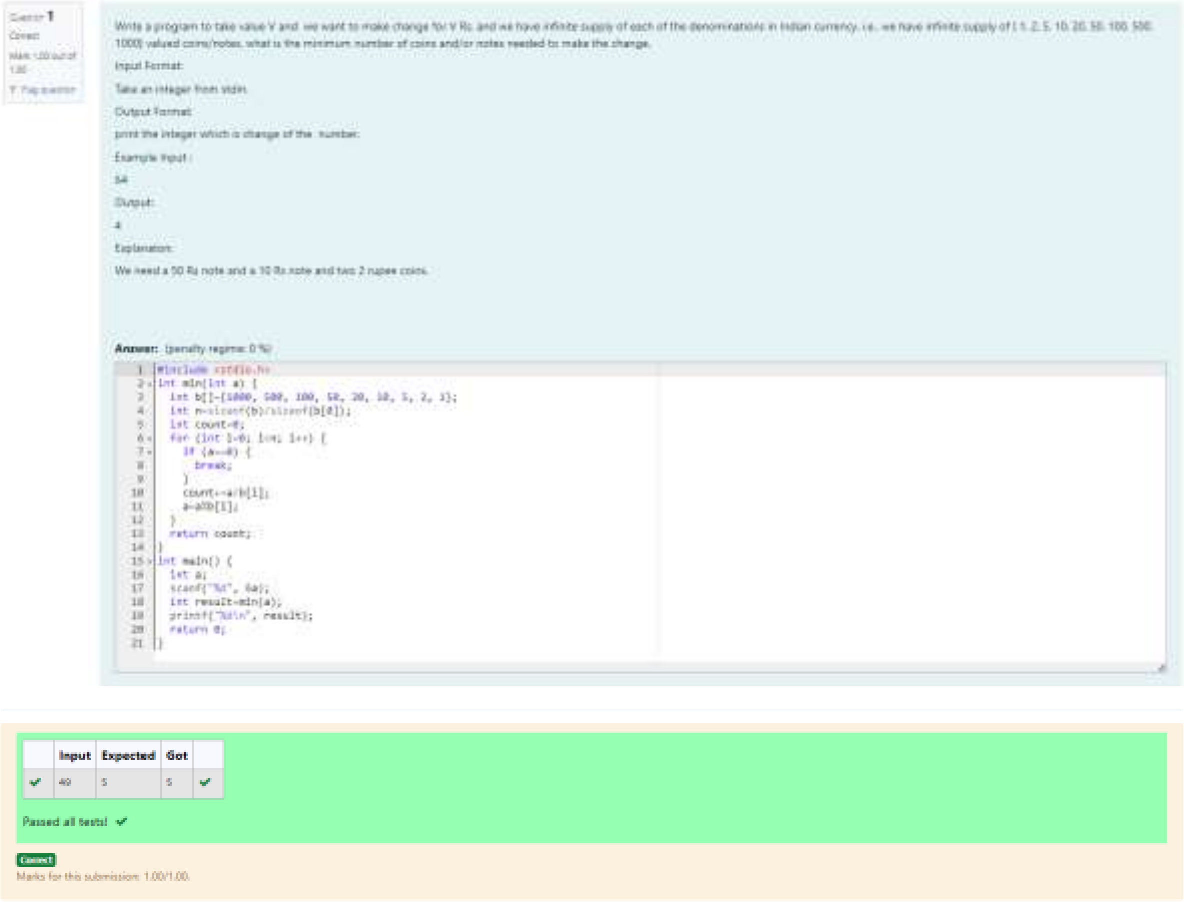
	Input	Expected	Got	
✓	12	11	11	✓
✓	1234	19	19	✓

Passed all tests! ✓

Marks for this submission: 1.00/1.00.

GREEDY METHOD

1.



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Write a program to take value V and we want to make change for V Rs. and we have infinite supply of each of the denominations in Indian currency, i.e., we have infinite supply of { 1, 2, 5, 10, 20, 50, 100, 500, 1000} valued coins/notes, what is the minimum number of coins and/or notes needed to make the change.

Input Format:

Take an integer from stdin.

Output Format:

print the integer which is change of the number.

Example Input :

64

Output:

4

Explanation:

We need a 50 Rs note and a 10 Rs note and two 2 rupee coins.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int min(int a) {
  int b[]={1000, 500, 100, 50, 20, 10, 5, 2, 1};
  int n=sizeof(b)/sizeof(b[0]);
  int count=0;
  for (int i=0; i<n; i++) {
    if (a==0) {
      break;
    }
    count+=a/b[i];
    a=a%b[i];
  }
  return count;
}
int main() {
  int a;
  scanf("%d", &a);
  int result=min(a);
  printf("%d\n", result);
  return 0;
}
```

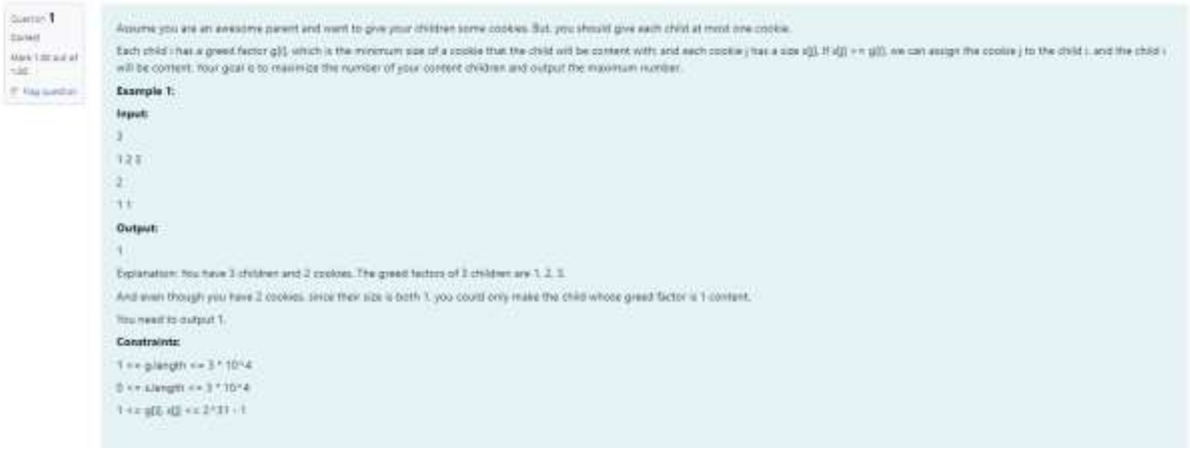
	Input	Expected	Got	
✓	49	5	5	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

2.Cookies Problem



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.

Each child i has a greed factor g[i], which is the minimum size of a cookie that the child will be content with; and each cookie j has a size s[j]. If s[j] >= g[i], we can assign the cookie j to the child i, and the child i will be content. Your goal is to maximize the number of your content children and output the maximum number.

Example 1:

Input:

3

1 2 3

2

1 1

Output:

1

Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1, 2, 3.

And even though you have 2 cookies, since their size is both 1, you could only make the child whose greed factor is 1 content.

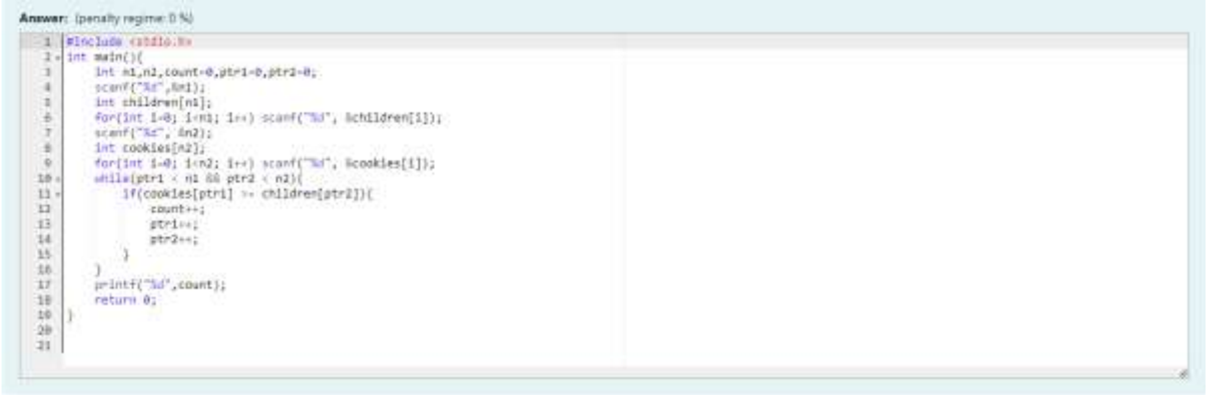
You need to output 1.

Constraints:

1 <= g.length <= 3 * 10^4

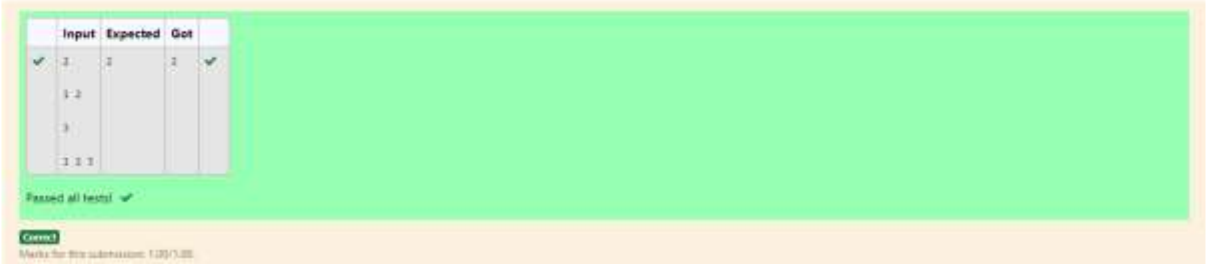
0 <= s.length <= 3 * 10^4

1 <= g[i], s[j] <= 2^31 - 1



Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int main(){
    int n1,n2,count=0,ptr1=0,ptr2=0;
    scanf("%d",&n1);
    int children[n1];
    for(int i=0; i<n1; i++) scanf("%d", &children[i]);
    scanf("%d", &n2);
    int cookies[n2];
    for(int i=0; i<n2; i++) scanf("%d", &cookies[i]);
    while(ptr1 < n1 && ptr2 < n2){
        if(cookies[ptr1] >= children[ptr2]){
            count++;
            ptr1++;
            ptr2++;
        }
    }
    printf("%d",count);
    return 0;
}
```



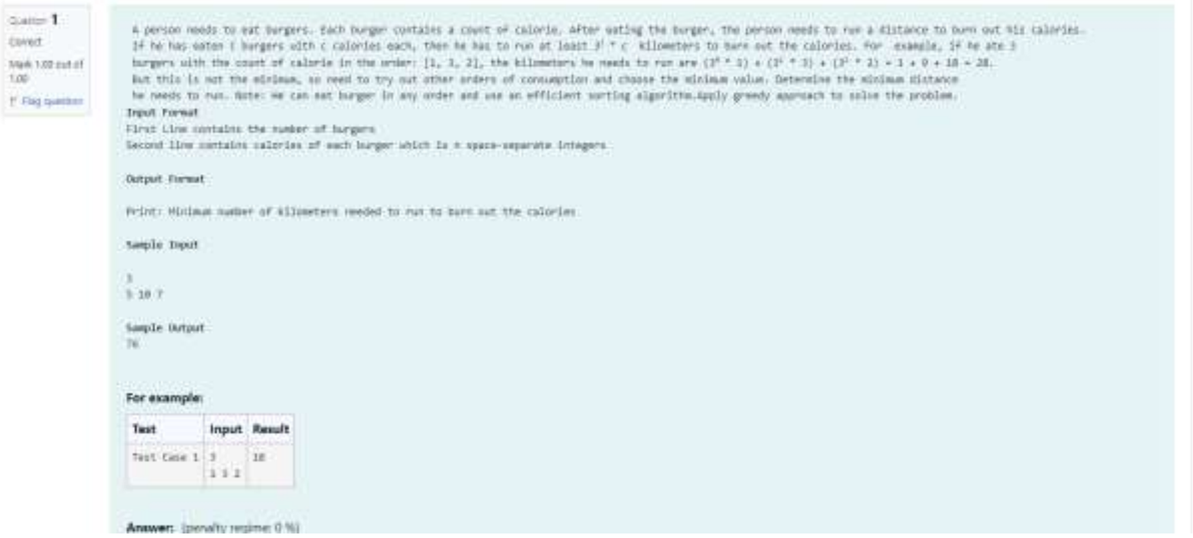
	Input	Expected	Got	
✓	2 1 2 3 1 2 3	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

3.Burger Problem



A person needs to eat burgers. Each burger contains a count of calorie. After eating the burger, the person needs to run a distance to burn out his calories. If he has eaten i burgers with c calories each, then he has to run at least $3^i * c$ kilometers to burn out the calories. For example, if he ate 3 burgers with the count of calorie in the order: [1, 3, 2], the kilometers he needs to run are $(3^0 * 1) + (3^1 * 3) + (3^2 * 2) = 1 + 9 + 18 = 28$.
But this is not the minimum, so need to try out other orders of consumption and choose the minimum value. Determine the minimum distance he needs to run. Note: He can eat burger in any order and use an efficient sorting algorithm.Apply greedy approach to solve the problem.

Input Format

First Line contains the number of burgers

Second line contains calories of each burger which is n space-separate integers

Output Format

Print: Minimum number of kilometers needed to run to burn out the calories

Sample Input

```
3
5 10 7
```

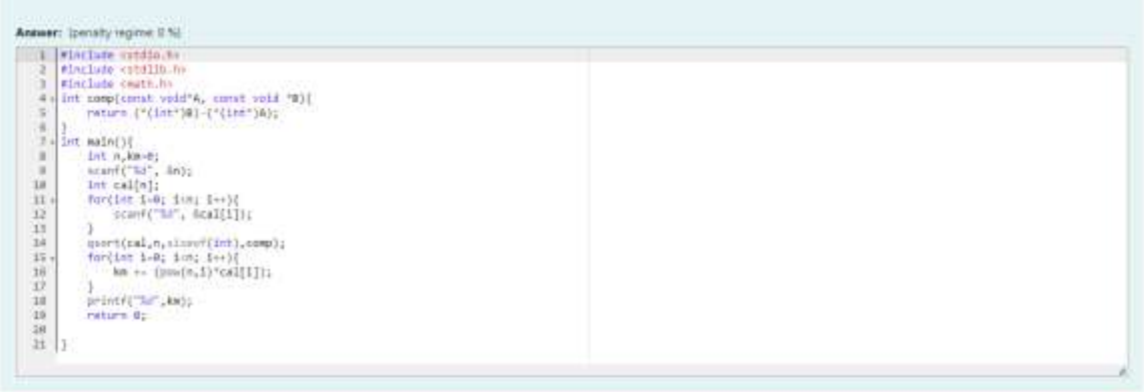
Sample Output

```
76
```

For example:

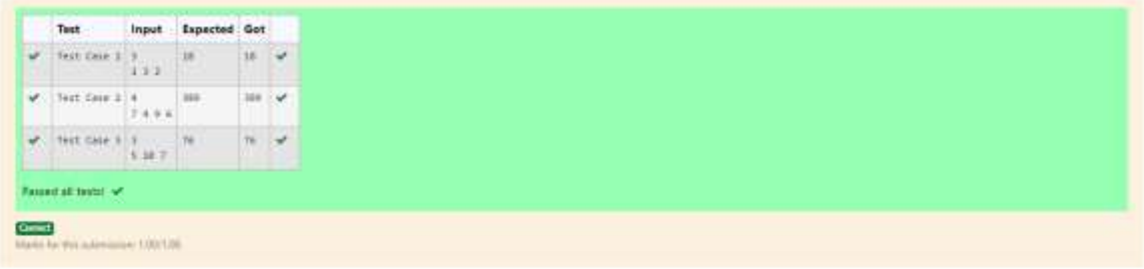
Test	Input	Result
Test Case 1	3 1 3 2	18

Answer: (penalty regime: 0 %)



Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
int comp(const void*A, const void *B){
    return (*(int*)B)-(*(int*)A);
}
int main(){
    int n,km=0;
    scanf("%d", &n);
    int cal[n];
    for(int i=0; i<n; i++){
        scanf("%d", &cal[i]);
    }
    qsort(cal,n,sizeof(int),comp);
    for(int i=0; i<n; i++){
        km += (pow(n,i)*cal[i]);
    }
    printf("%d",km);
    return 0;

}
```

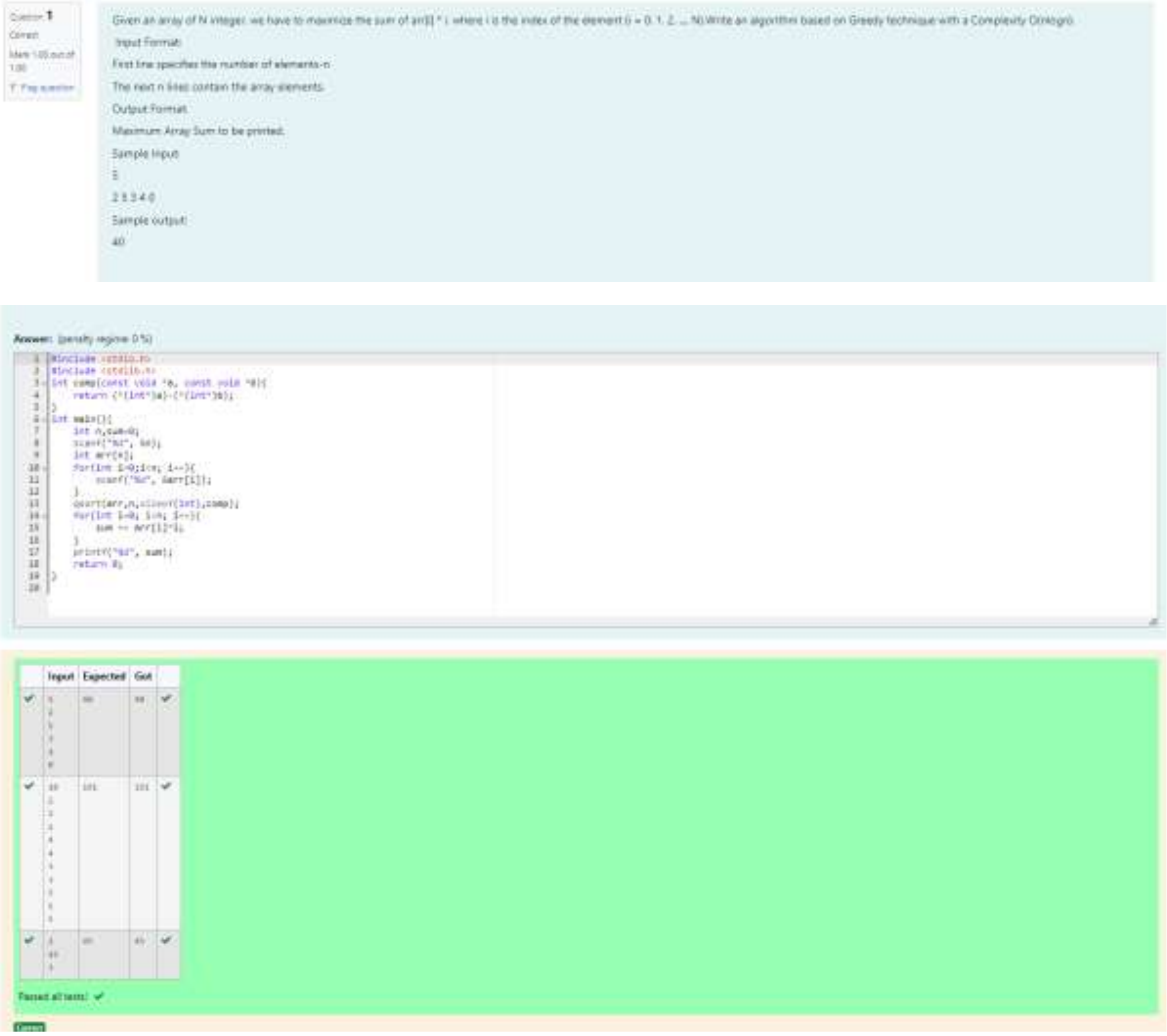


	Test	Input	Expected	Got	
✓	Test Case 1	3 1 3 2	18	18	✓
✓	Test Case 2	4 7 4 9 6	389	389	✓
✓	Test Case 3	3 5 10 7	76	76	✓

Passed all tests! ✓

Marks for this submission: 1.00/1.00

4.Array sum Max problem



Question **1**

Correct

Mark 1.00 out of 1.00

Flag question

Given an array of N integer, we have to maximize the sum of arr[i] * i, where i is the index of the element (i = 0, 1, 2, ..., N).Write an algorithm based on Greedy technique with a Complexity O(nlogn).

Input Format:

First line specifies the number of elements-n

The next n lines contain the array elements.

Output Format:

Maximum Array Sum to be printed.

Sample Input:

5

2 5 3 4 0

Sample output:

40

Answer: (penalty regime: 0 %)

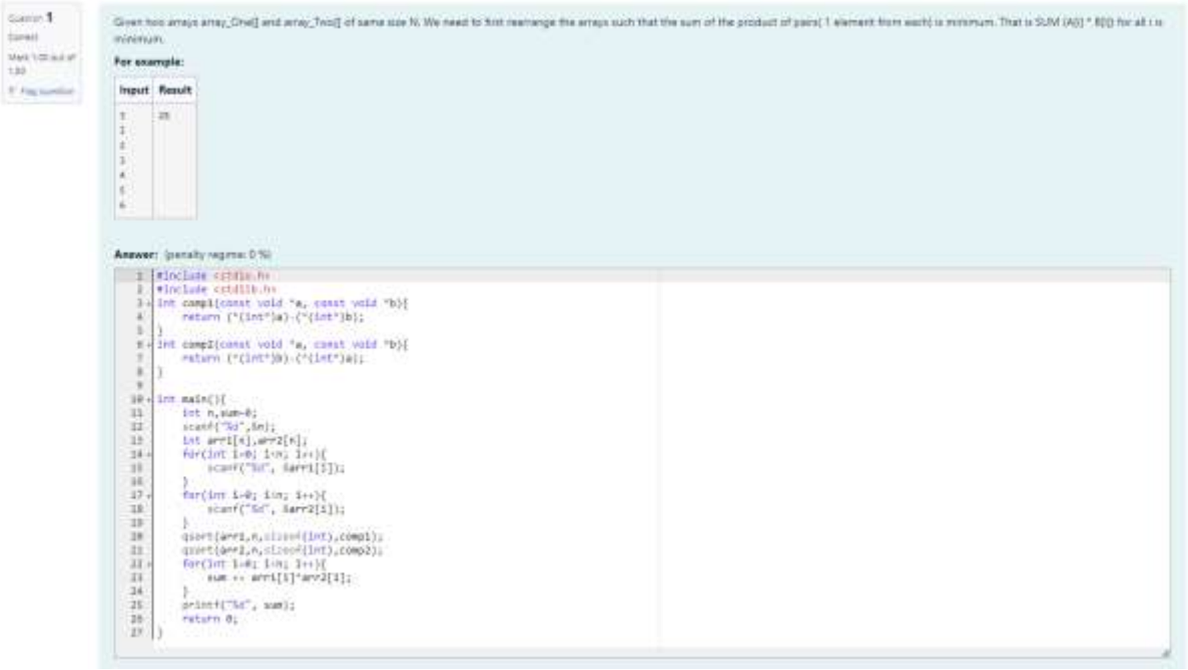
```c
#include <stdio.h>
#include <stdlib.h>
int comp(const void *a, const void *b){
    return (*(int*)a)-(*(int*)b);
}
int main(){
    int n,sum=0;
    scanf("%d", &n);
    int arr[n];
    for(int i=0;i<n; i++){
        scanf("%d", &arr[i]);
    }
    qsort(arr,n,sizeof(int),comp);
    for(int i=0; i<n; i++){
        sum += arr[i]*i;
    }
    printf("%d", sum);
    return 0;
}
```

	Input	Expected	Got	
✓	5 2 5 3 4 0	40	40	✓
✓	10 2 2 2 4 4 3 3 5 5 5	191	191	✓
✓	2 45 3	45	45	✓

Passed all tests! ✓

Correct

5. GREEDY METHOD



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

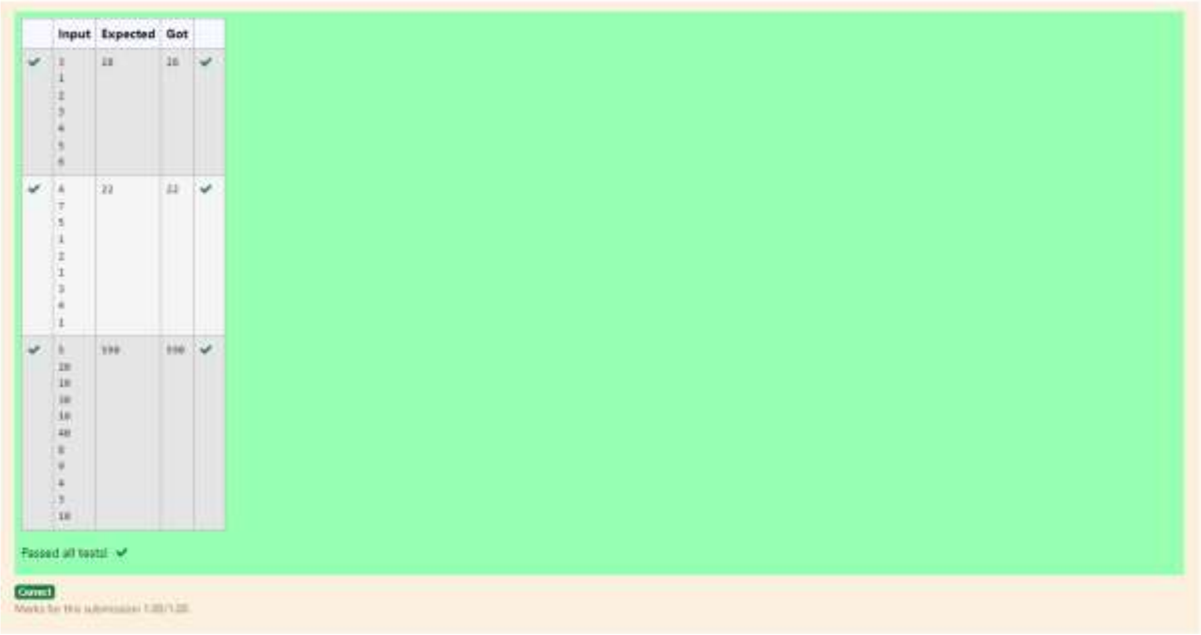
Given two arrays array_One[] and array_Two[] of same size N. We need to first rearrange the arrays such that the sum of the product of pairs(1 element from each) is minimum. That is SUM (A[i] * B[i]) for all i is minimum.

For example:

Input	Result
3 1 2 3 4 5 6	28

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <stdlib.h>
int comp1(const void *a, const void *b){
    return (*(int*)a)-(*(int*)b);
}
int comp2(const void *a, const void *b){
    return (*(int*)b)-(*(int*)a);
}

int main(){
    int n,sum=0;
    scanf("%d",&n);
    int arr1[n],arr2[n];
    for(int i=0; i<n; i++){
        scanf("%d", &arr1[i]);
    }
    for(int i=0; i<n; i++){
        scanf("%d", &arr2[i]);
    }
    qsort(arr1,n,sizeof(int),comp1);
    qsort(arr2,n,sizeof(int),comp2);
    for(int i=0; i<n; i++){
        sum += arr1[i]*arr2[i];
    }
    printf("%d", sum);
    return 0;
}
```



	Input	Expected	Got	
✓	3 1 2 3 4 5 6	28	28	✓
✓	4 7 5 1 2 1 3 4 1	22	22	✓
✓	5 20 10 30 10 40 8 9 4 3 10	590	590	✓

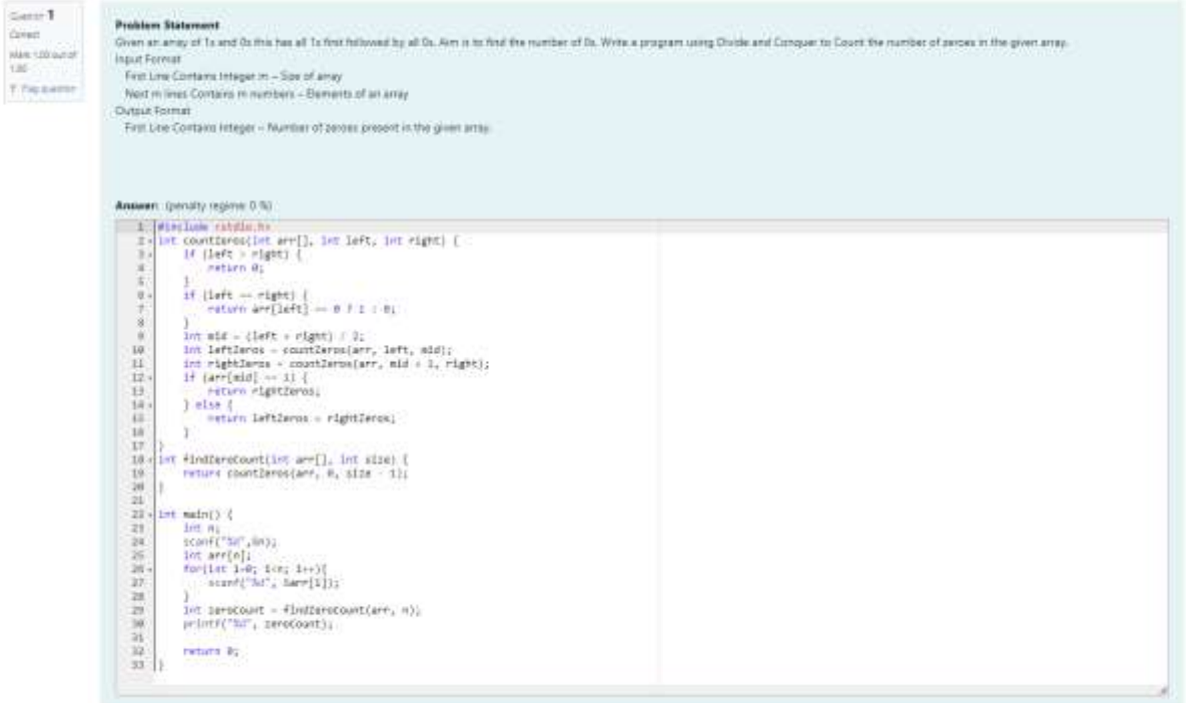
Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

DIVIDE AND CONQUER

1.



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Problem Statement

Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program using Divide and Conquer to Count the number of zeroes in the given array.

Input Format

First Line Contains Integer m – Size of array

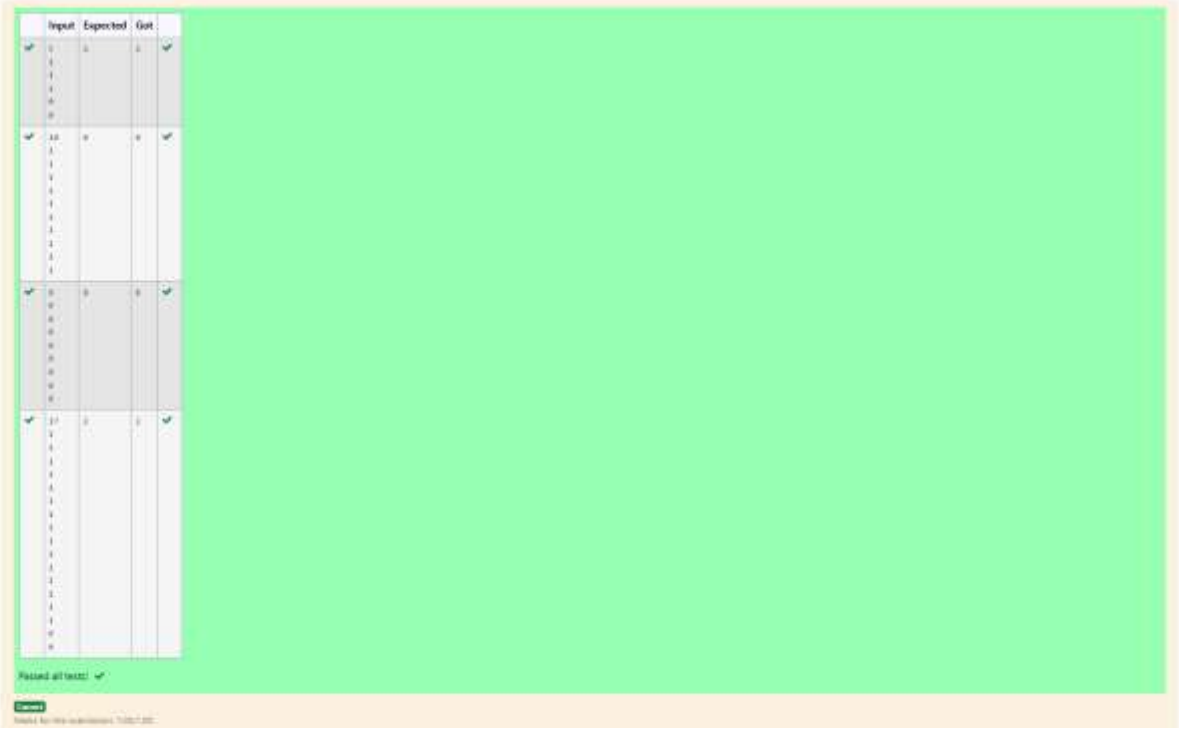
Next m lines Contains m numbers – Elements of an array

Output Format

First Line Contains Integer – Number of zeroes present in the given array.

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
int countZeros(int arr[], int left, int right) {
    if (left > right) {
        return 0;
    }
    if (left == right) {
        return arr[left] == 0 ? 1 : 0;
    }
    int mid = (left + right) / 2;
    int leftZeros = countZeros(arr, left, mid);
    int rightZeros = countZeros(arr, mid + 1, right);
    if (arr[mid] == 1) {
        return rightZeros;
    } else {
        return leftZeros + rightZeros;
    }
}
int findZeroCount(int arr[], int size) {
    return countZeros(arr, 0, size - 1);
}

int main() {
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0; i<n; i++){
        scanf("%d", &arr[i]);
    }
    int zeroCount = findZeroCount(arr, n);
    printf("%d", zeroCount);

    return 0;
}
```



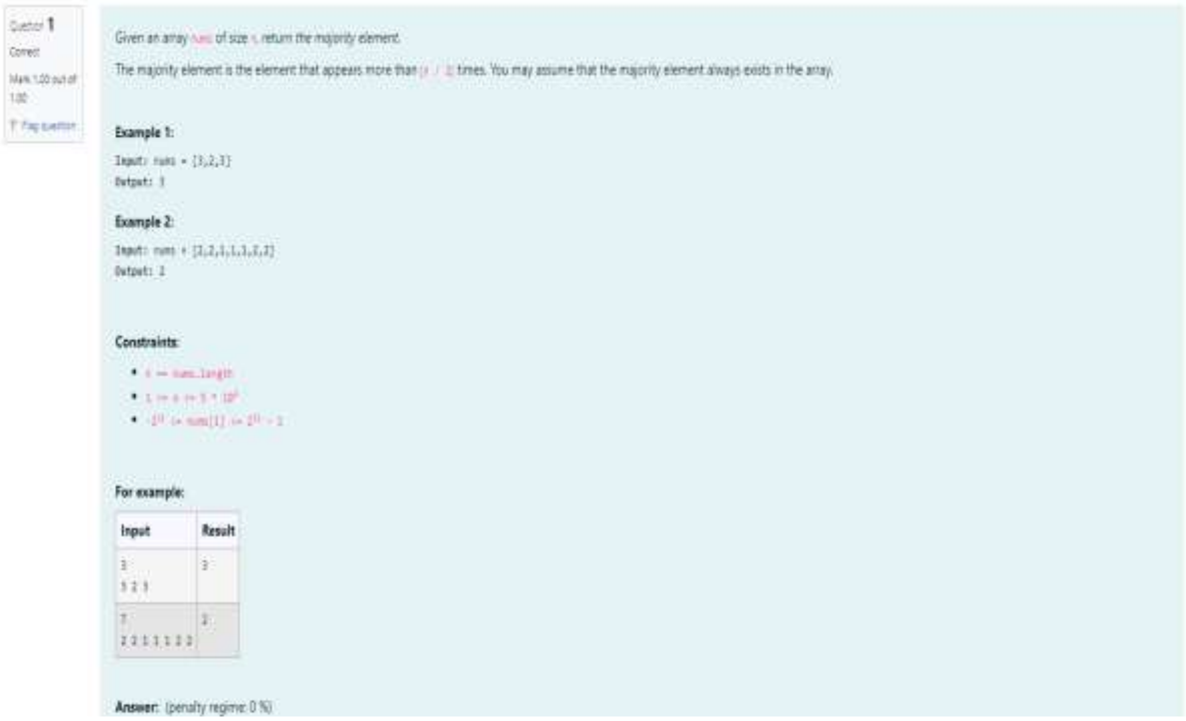
	Input	Expected	Got	
✓	5 1 1 1 0 0	2	2	✓
✓	10 1 1 1 1 1 1 1 1 1 1	0	0	✓
✓	8 0 0 0 0 0 0 0 0	8	8	✓
✓	17 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

2. DIVIDE AND CONQUER



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Given an array nums of size n, return *the majority element*.

The majority element is the element that appears more than ⌊n / 2⌋ times. You may assume that the majority element always exists in the array.

Example 1:

```
Input: nums = [3,2,3]
Output: 3
```

Example 2:

```
Input: nums = [2,2,1,1,1,2,2]
Output: 2
```

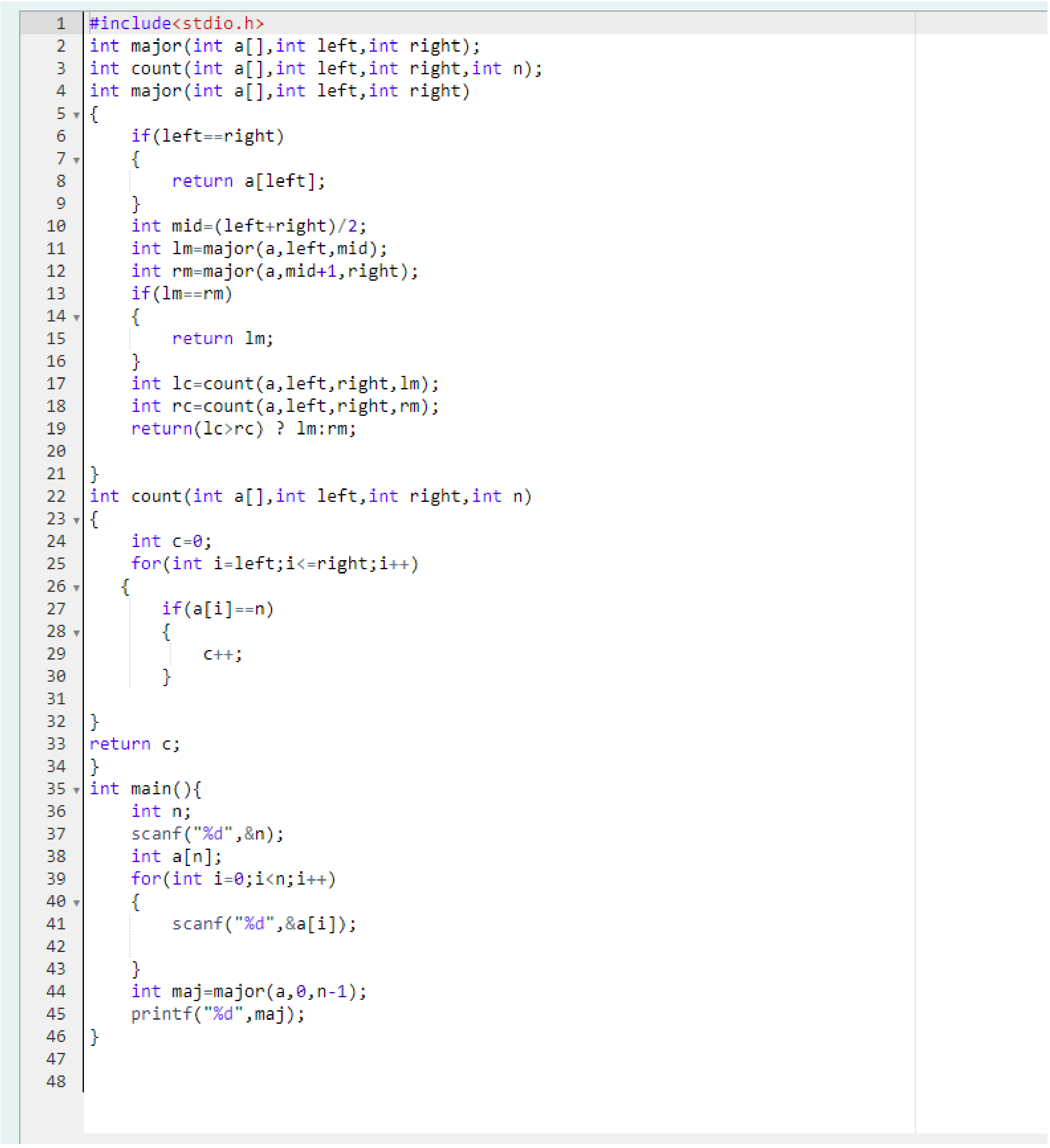
Constraints:

- `n == nums.length`
- `1 <= n <= 5 * 10`^{4}
- `-2`^{31} `<= nums[i] <= 2`^{31} `- 1`

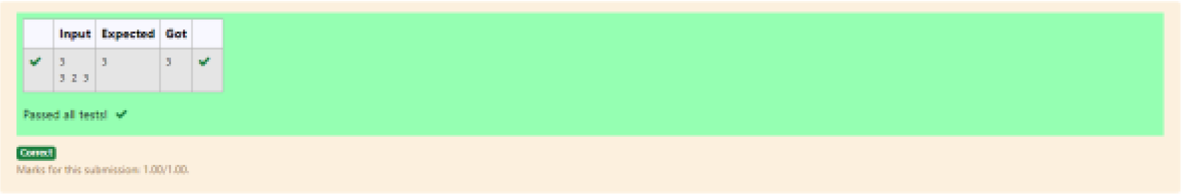
For example:

Input	Result
3 3 2 3	3
7 2 2 1 1 1 2 2	2

Answer: (penalty regime: 0 %)



```c
#include<stdio.h>
int major(int a[],int left,int right);
int count(int a[],int left,int right,int n);
int major(int a[],int left,int right)
{
    if(left==right)
    {
        return a[left];
    }
    int mid=(left+right)/2;
    int lm=major(a,left,mid);
    int rm=major(a,mid+1,right);
    if(lm==rm)
    {
        return lm;
    }
    int lc=count(a,left,right,lm);
    int rc=count(a,left,right,rm);
    return(lc>rc) ? lm:rm;

}
int count(int a[],int left,int right,int n)
{
    int c=0;
    for(int i=left;i<=right;i++)
   {
       if(a[i]==n)
       {
           c++;
       }

}
return c;
}
int main(){
    int n;
    scanf("%d",&n);
    int a[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&a[i]);

    }
    int maj=major(a,0,n-1);
    printf("%d",maj);
}


```



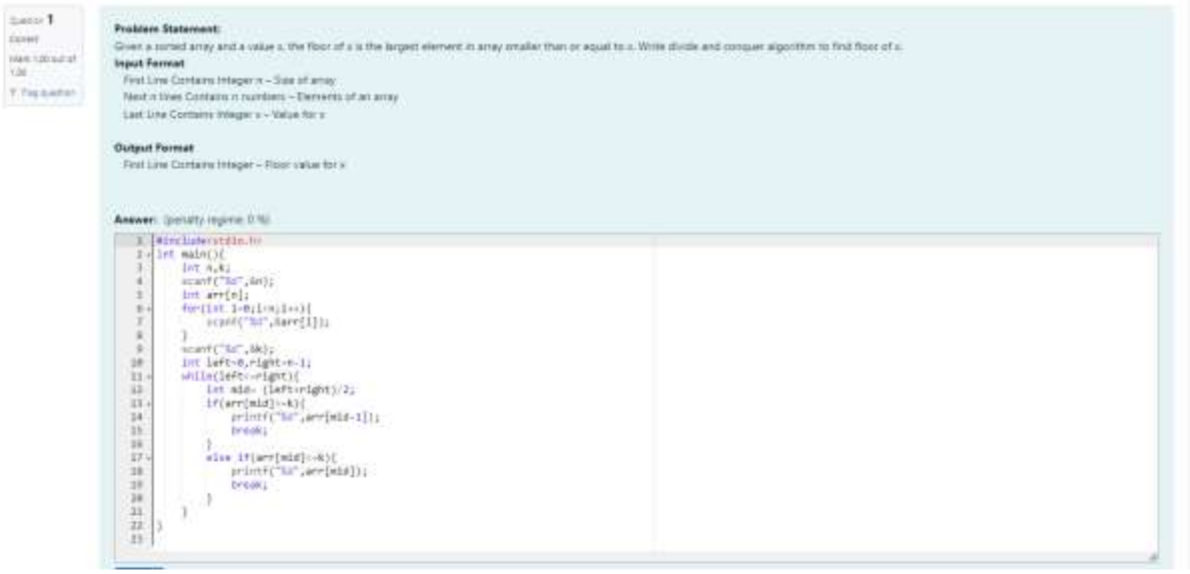
	Input	Expected	Got	
✓	3 3 2 3	3	3	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

3. DIVIDE AND CONQUER



Question **1**

Correct

Mark 1.00 out of 1.00

Flag question

Problem Statement:

Given a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

Input Format

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

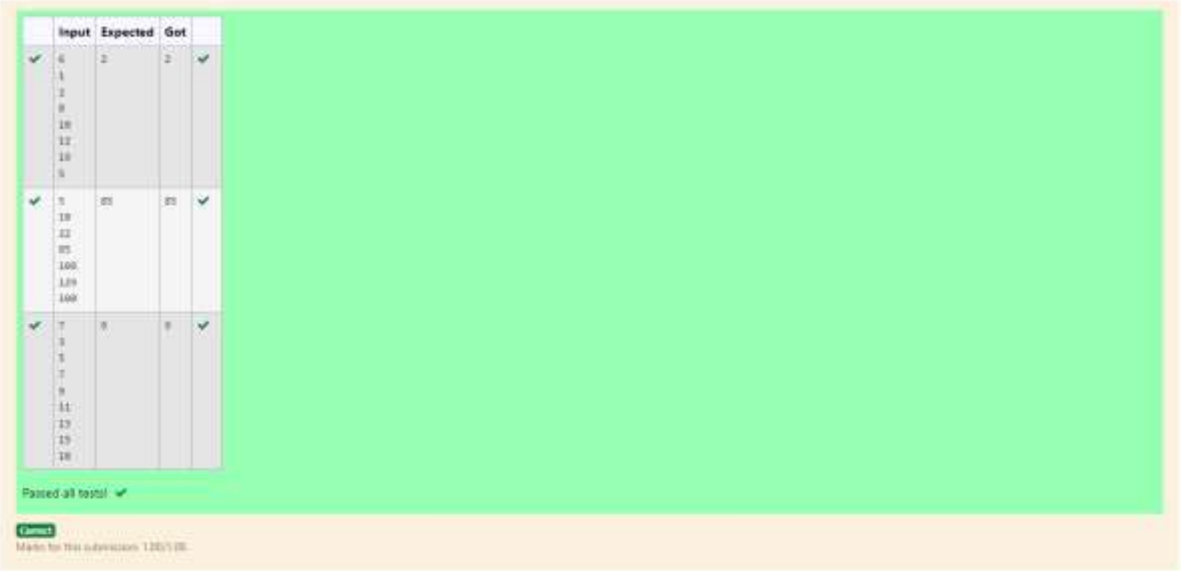
Last Line Contains Integer x – Value for x

Output Format

First Line Contains Integer – Floor value for x

Answer: (penalty regime: 0 %)

```c
#include<stdio.h>
int main(){
    int n,k;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    scanf("%d",&k);
    int left=0,right=n-1;
    while(left<=right){
        int mid= (left+right)/2;
        if(arr[mid]>k){
            printf("%d",arr[mid-1]);
            break;
        }
        else if(arr[mid]<=k){
            printf("%d",arr[mid]);
            break;
        }
    }
}

```



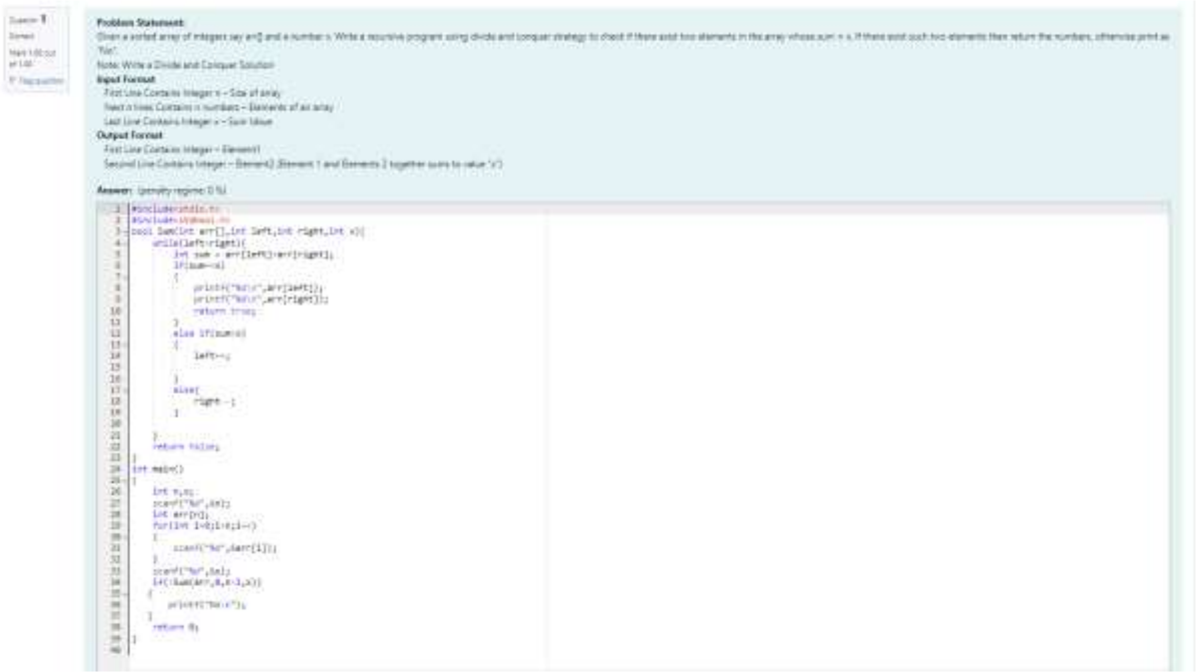
	Input	Expected	Got	
✓	6 1 2 8 10 12 19 5	2	2	✓
✓	5 10 22 85 108 129 100	85	85	✓
✓	7 3 5 7 9 11 13 15 10	9	9	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

4.DIVIDE AND CONQUER



Question 1
Correct
Mark 1.00 out of 1.00
Flag question

Problem Statement:

Given a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return the numbers, otherwise print as "No".

Note: Write a Divide and Conquer Solution

Input Format

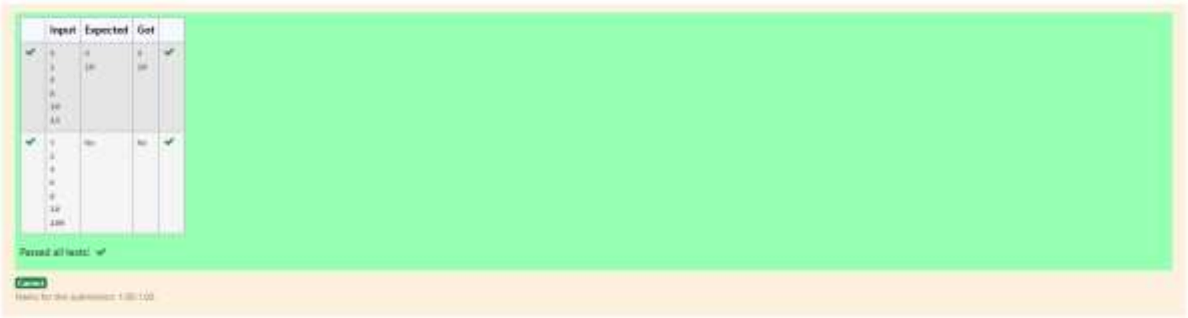
First Line Contains Integer n – Size of array
Next n lines Contains n numbers – Elements of an array
Last Line Contains Integer x – Sum Value

Output Format

First Line Contains Integer – Element1
Second Line Contains Integer – Element2 (Element 1 and Elements 2 together sums to value "x")

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
#include<stdbool.h>
bool Sum(int arr[],int left,int right,int x){
    while(left<right){
        int sum = arr[left]+arr[right];
        if(sum==x)
        {
            printf("%d\n",arr[left]);
            printf("%d\n",arr[right]);
            return true;
        }
        else if(sum<x)
        {
            left++;

        }
        else{
            right--;
        }

    }
    return false;
}
int main()
{
    int n,x;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    scanf("%d",&x);
    if(!Sum(arr,0,n-1,x))
    {
        printf("No\n");
    }
    return 0;
}
```

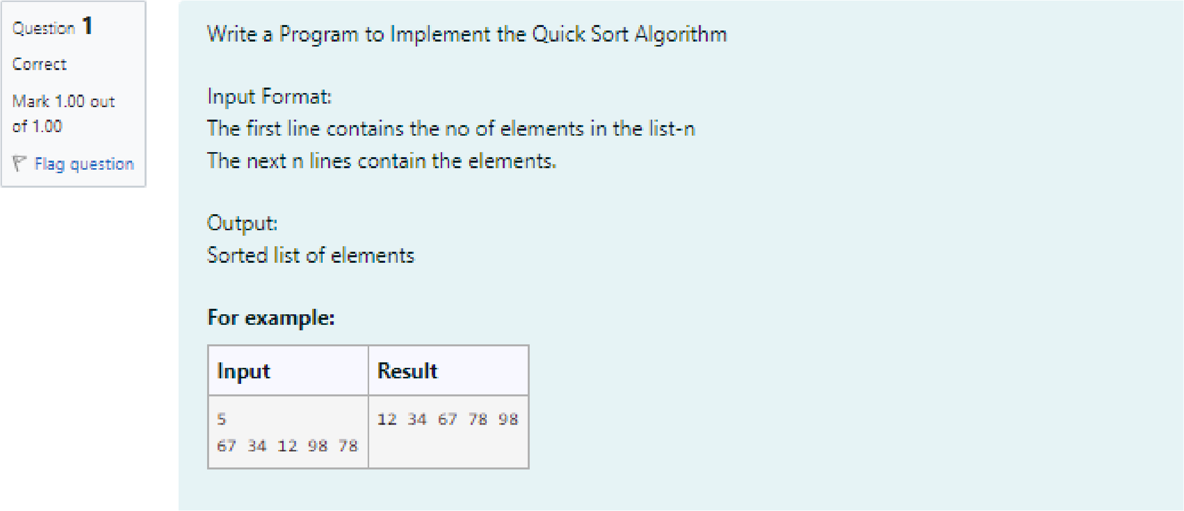


	Input	Expected	Got	
✓	4 2 4 8 10 14	4 10	4 10	✓
✓	5 2 4 6 8 10 100	No	No	✓

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.

5. DIVIDE AND CONQUER



Question **1**

Correct

Mark 1.00 out of 1.00

Flag question

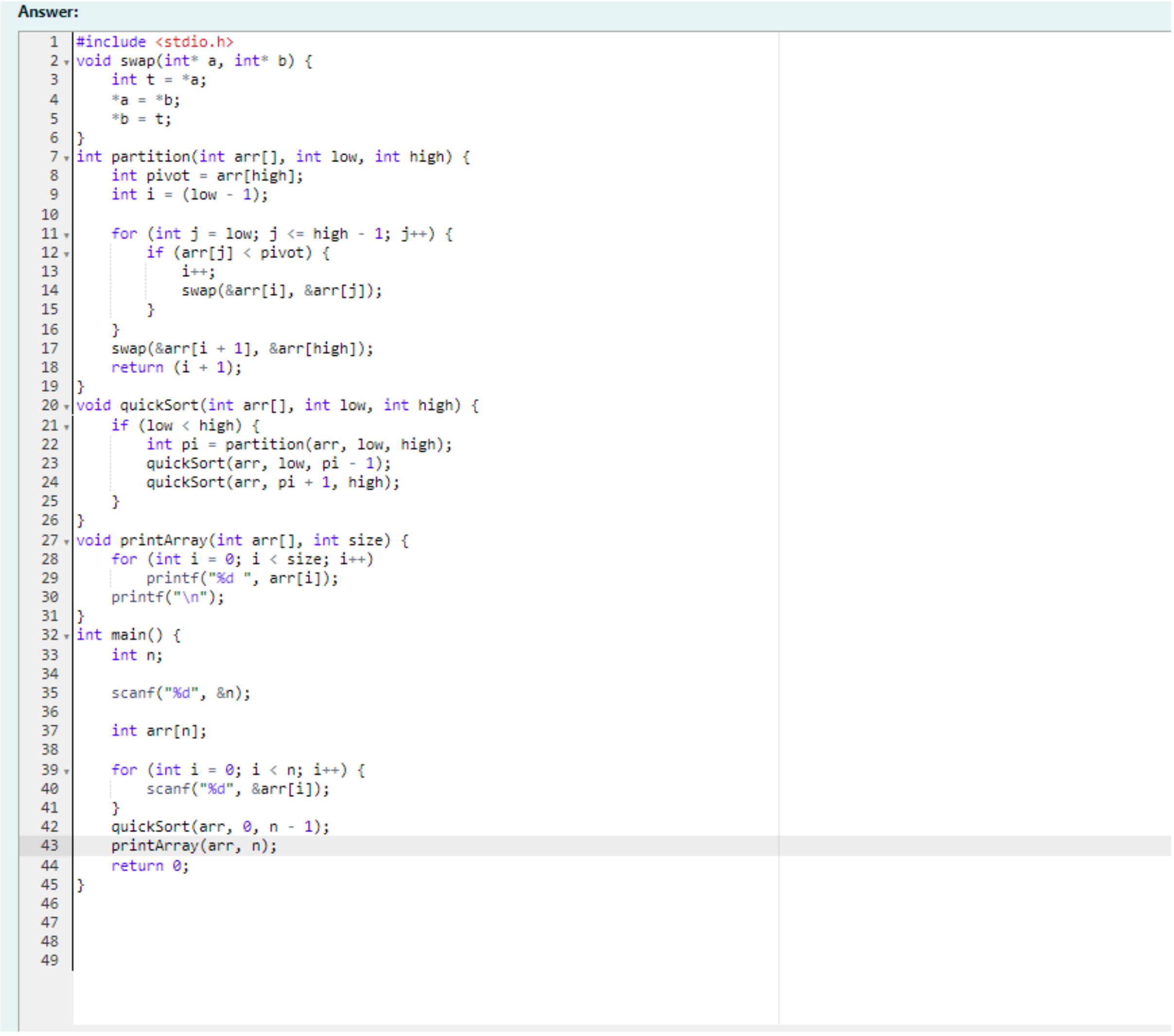
Write a Program to Implement the Quick Sort Algorithm

Input Format:
The first line contains the no of elements in the list-n
The next n lines contain the elements.

Output:
Sorted list of elements

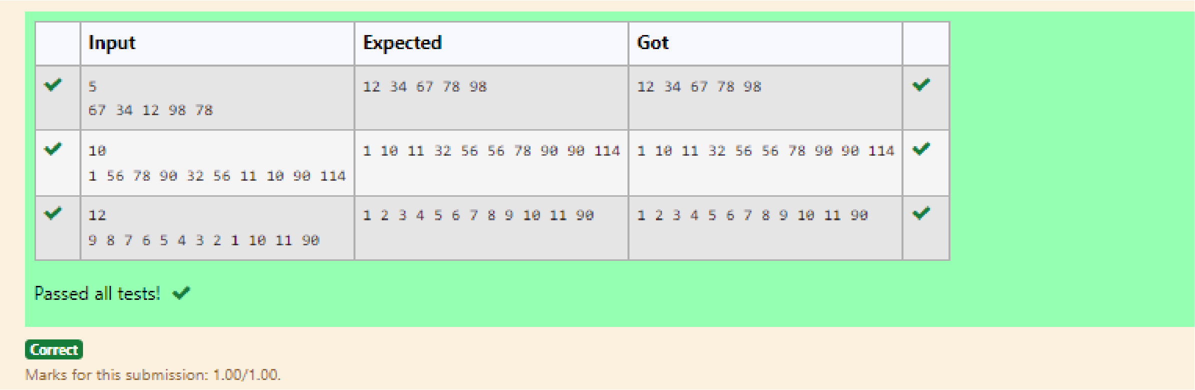
For example:

Input	Result
5 67 34 12 98 78	12 34 67 78 98



Answer:

```
#include <stdio.h>
void swap(int* a, int* b) {
    int t = *a;
    *a = *b;
    *b = t;
}
int partition(int arr[], int low, int high) {
    int pivot = arr[high];
    int i = (low - 1);

    for (int j = low; j <= high - 1; j++) {
        if (arr[j] < pivot) {
            i++;
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[i + 1], &arr[high]);
    return (i + 1);
}
void quickSort(int arr[], int low, int high) {
    if (low < high) {
        int pi = partition(arr, low, high);
        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}
void printArray(int arr[], int size) {
    for (int i = 0; i < size; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
int main() {
    int n;

    scanf("%d", &n);

    int arr[n];

    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    quickSort(arr, 0, n - 1);
    printArray(arr, n);
    return 0;
}
```



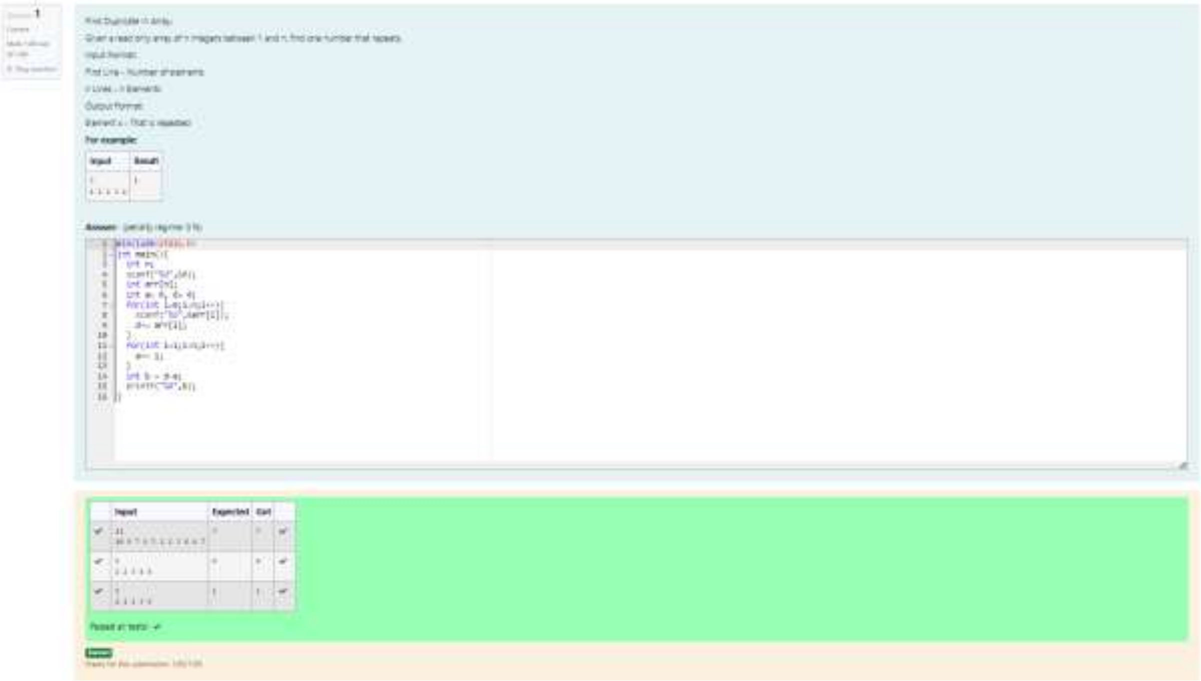
	Input	Expected	Got	
✓	5 67 34 12 98 78	12 34 67 78 98	12 34 67 78 98	✓
✓	10 1 56 78 90 32 56 11 10 90 114	1 10 11 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	✓
✓	12 9 8 7 6 5 4 3 2 1 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

COMPETITIVE PROGRAMMING



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5 1 1 2 3 4	1

Answer: (penalty regime: 0 %)

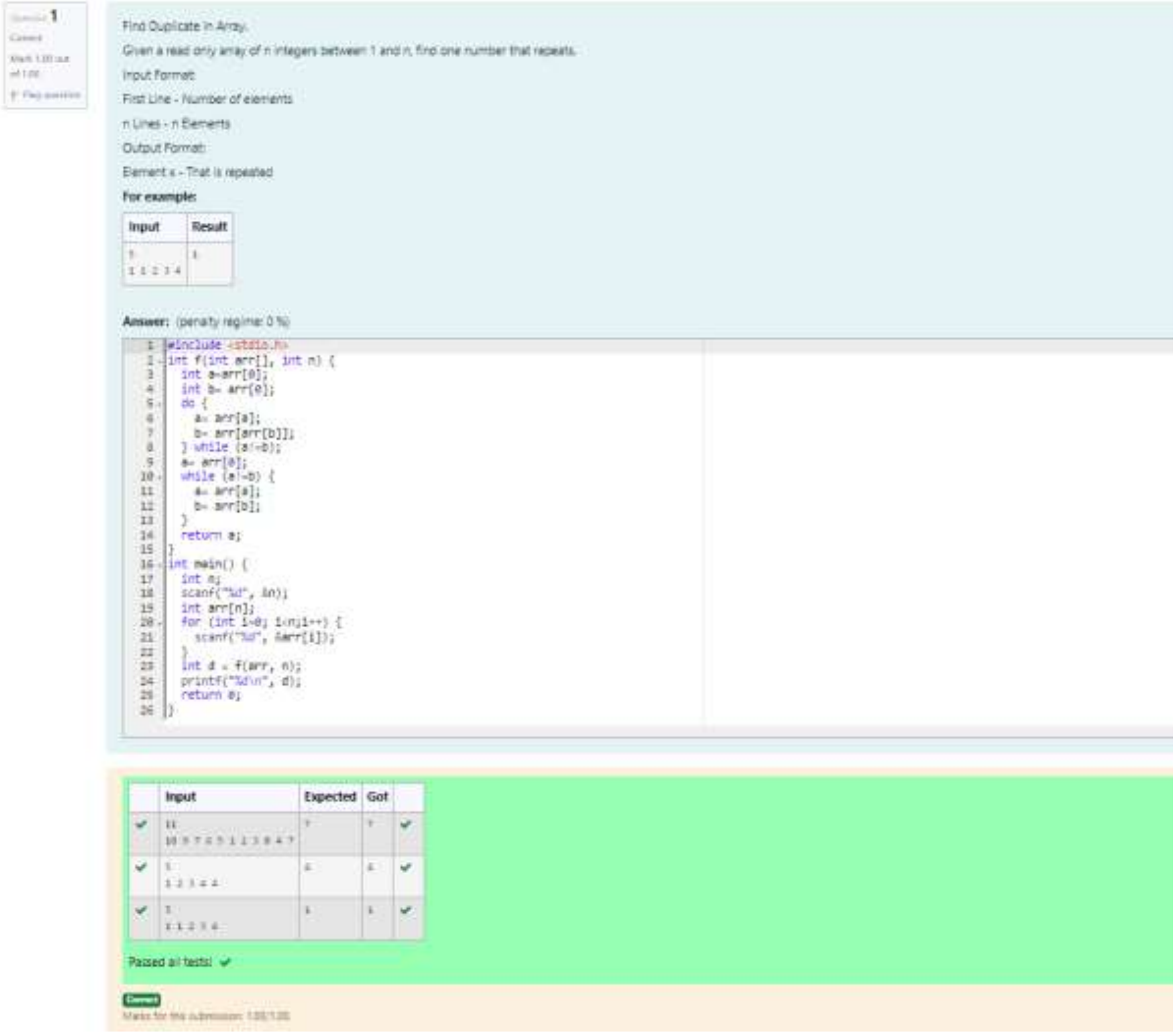
```
#include<stdio.h>
int main(){
  int n;
  scanf("%d",&n);
  int arr[n];
  int a= 0, d= 0;
  for(int i=0;i<n;i++){
    scanf("%d",&arr[i]);
    a+= arr[i];
  }
  for(int i=1;i<n;i++){
    d+= i;
  }
  int b = a-d;
  printf("%d",b);
}
```

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.



Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5 1 1 2 3 4	1

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int f(int arr[], int n) {
  int a=arr[0];
  int b= arr[0];
  do {
    a= arr[a];
    b= arr[arr[b]];
  } while (a!=b);
  a= arr[0];
  while (a!=b) {
    a= arr[a];
    b= arr[b];
  }
  return a;
}
int main() {
  int n;
  scanf("%d", &n);
  int arr[n];
  for (int i=0; i<n;i++) {
    scanf("%d", &arr[i]);
  }
  int d = f(arr, n);
  printf("%d\n", d);
  return 0;
}
```

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.