NAME:KAVYASHREE S USN:4AL17IS022 DATE:05.08.19

# "Hello World!" in C

**Task**

This challenge requires you to print “Hello world”  on a single line, and then print the already provided input string to [stdout](https://en.wikipedia.org/wiki/Standard_streams" \l "Standard_output_.28stdout.29).

Note: You do not need to read any input in this challenge.

**Input Format**

You do not need to read any input in this challenge.

**Output Format**

Print  on the first line, and the string from the given input on the second line.

**Sample Input 0**

Welcome to C programming.

**Sample Output 0**

Hello, World!

Welcome to C programming.

**SOLUTION:**

#include <stdio.h>

#include<string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

char stdout[100];

char s[100];

strcpy(stdout,"Welcome to C programming.");

printf("Hello, World!");

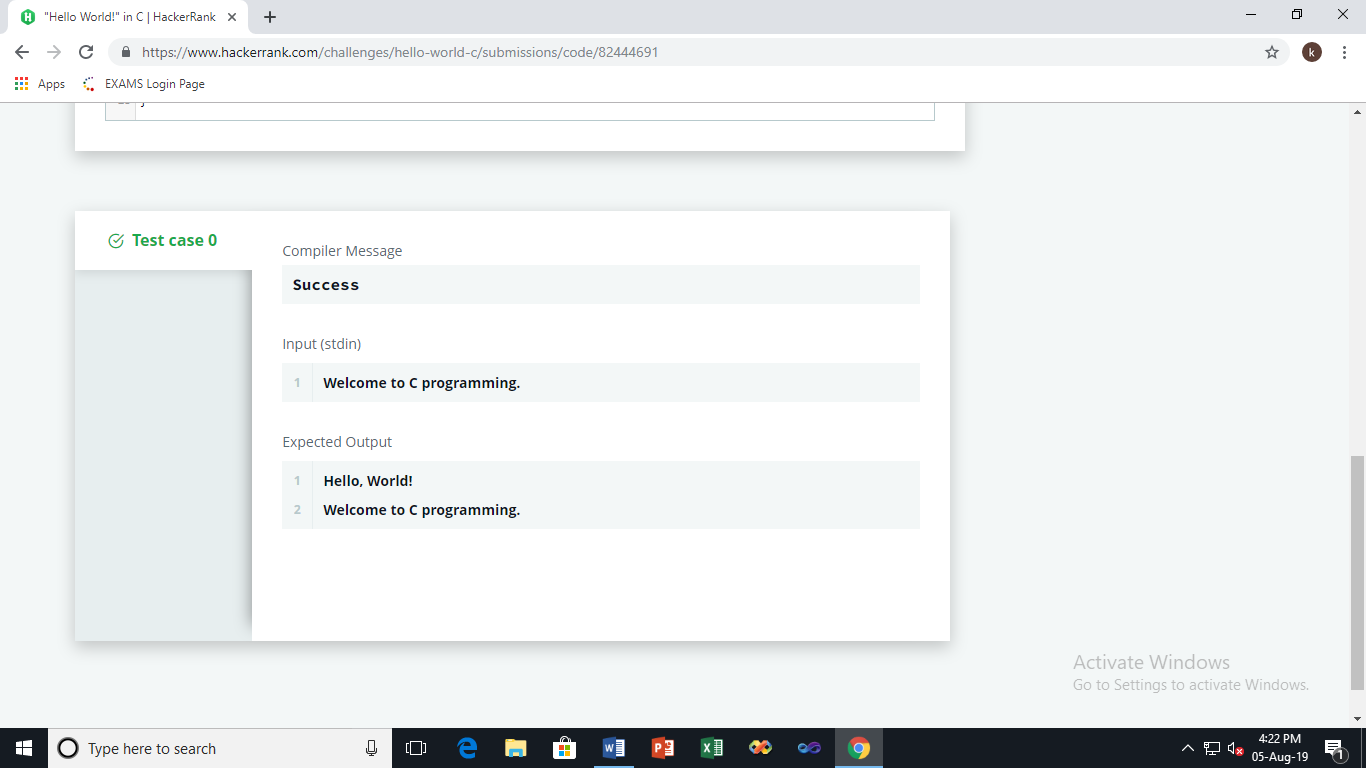
/\* Enter your code here. Read input from STDIN. Print output to STDOUT \*/

printf("\n%s",stdout);

return 0;

}

Output:



Score:05

# 2.Playing With Characters

**Task**

You have to print the character, , in the first line. Then print  in next line. In the last line print the sentence, .

**Input Format**

First, take a character,  as input.   
Then take the string,  as input.   
Lastly, take the sentence  as input.

**Output Format**

Print three lines of output. The first line prints the character, .   
The second line prints the string, .   
The third line prints the sentence, .

**Sample Input 0**

C

Language

Welcome To C!!

**Sample Output 0**

C

Language

Welcome To C!!

**Solution:**

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

int BUFFSIZE = 256;

char c;

char str[BUFFSIZE];

char sen[BUFFSIZE];

scanf("%c ", &c);

scanf("%s ", str);

fgets(sen, BUFFSIZE, stdin);

printf("%c\n", c);

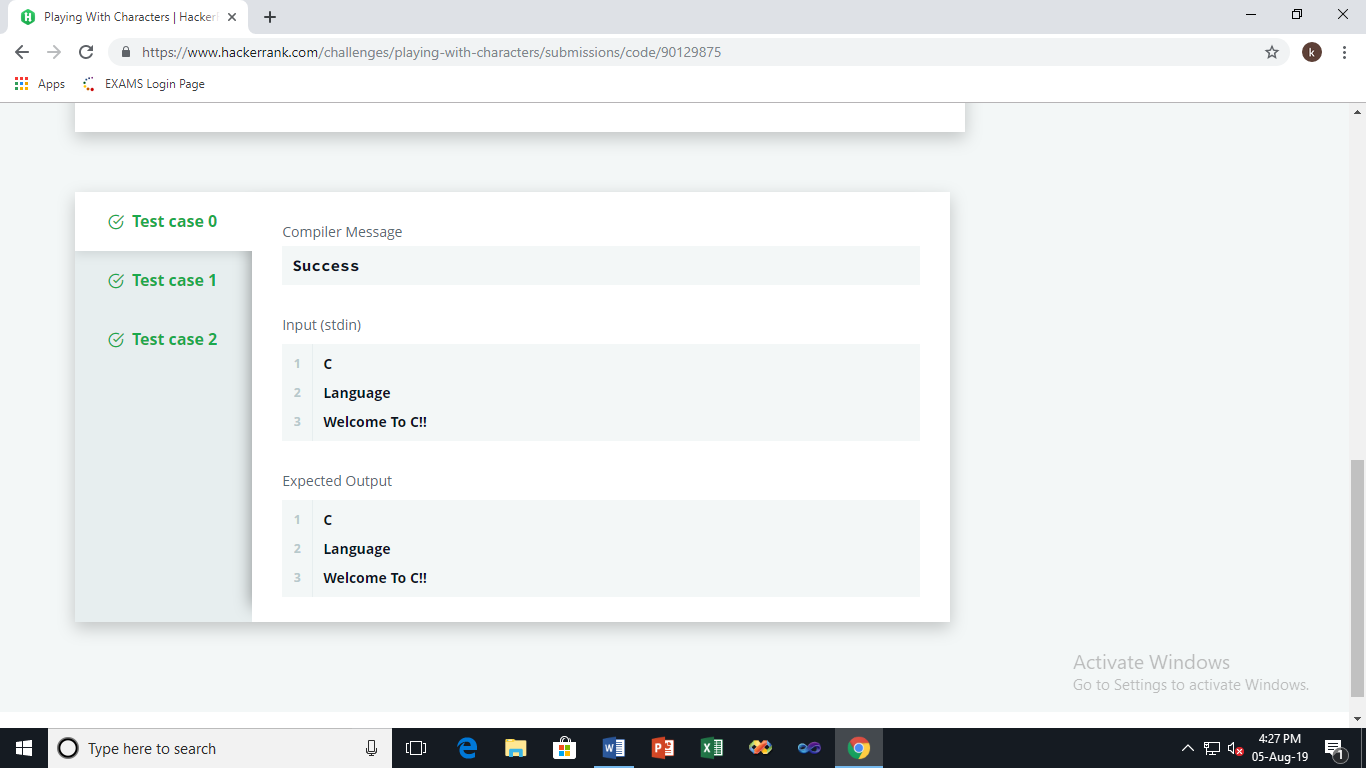
printf("%s\n", str);

printf("%s\n", sen);

return 0;

}

Output:



Score:05

# 3.Sum and Difference of Two Numbers

**Task**

Your task is to take two numbers of int data type, two numbers of float data type as input and output their sum:

1. Declare  variables: two of type int and two of type float.
2. Read  lines of input from stdin (according to the sequence given in the 'Input Format' section below) and initialize your variables.
3. Use the  and  operator to perform the following operations:
   * Print the sum and difference of two int variable on a new line.
   * Print the sum and difference of two float variable rounded to one decimal place on a new line.

**Input Format**

The first line contains two integers.   
The second line contains two floating point numbers.

**Constraints**

* integer variables
* float variables

**Output Format**

Print the sum and difference of both integers separated by a space on the first line, and the sum and difference of both float (scaled to  decimal place) separated by a space on the second line.

**Sample Input**

10 4

4.0 2.0

**Sample Output**

14 6

6.0 2.0

# Solution:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

int sum1,diff1;

float sum2,diff2;

int a,b;

float c,d;

scanf("%d%d",&a,&b);

sum1=a+b;

diff1=a-b;

scanf("%f%f",&c,&d);

sum2=c+d;

diff2=c-d;

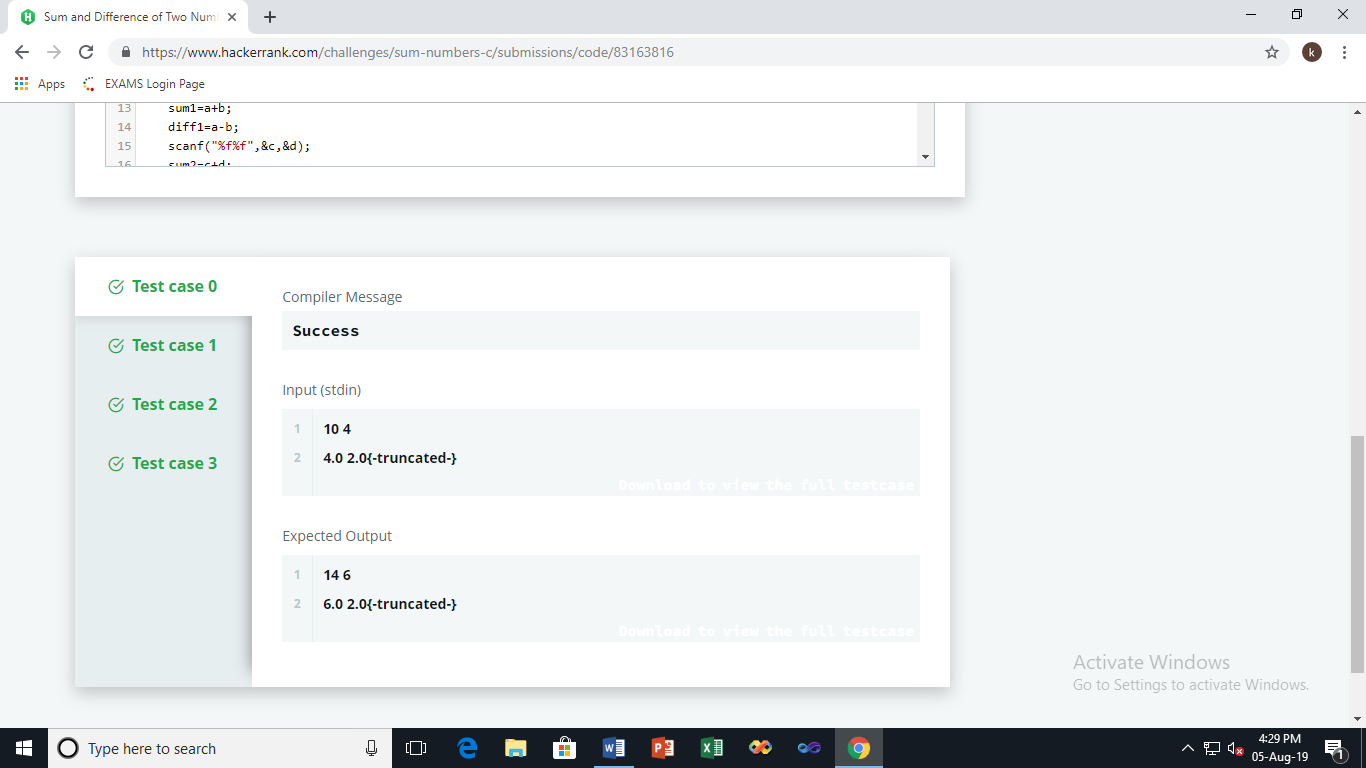
printf("%d %d\n",sum1,diff1);

printf("%.1f %.1f\n",sum2,diff2);

return 0;

}

Output:



Score: 5

# 4.Functions in C

**Task**

You have to write a function int max\_of\_four(int a, int b, int c, int d) which reads four arguments and returns the greatest of them.

+= : Add and assignment operator. It adds the right operand to the left operand and assigns the result to the left operand.

a += b is equivalent to a = a + b;

**Input Format**

Input will contain four integers -  , one in each line.

**Output Format**

Print the greatest of the four integers.   
Note: I/O will be automatically handled.

**Sample Input**

3

4

6

5

**Sample Output**

6

# Solution:

# #include <stdio.h>

# int max\_of\_four(int a,int b,int c,int d)

# {

# if((a>b)&&(a>c)&&(a>d))

# {

# return a;

# }

# else if((b>a)&&(b>c)&&(b>d))

# {

# return b;

# 

# }

# else if((c>a)&&(c>b)&&(c>d))

# {

# return c;

# }

# else

# return d;

# }

# /\*

# Add `int max\_of\_four(int a, int b, int c, int d)` here.

# \*/

# int main() {

# int a, b, c, d;

# scanf("%d %d %d %d", &a, &b, &c, &d);

# int ans = max\_of\_four(a, b, c, d);

# printf("%d", ans);

# 

# return 0;

# }

Output:

# 

Score: 10

# 5.Pointers in C

**Task**

You have to complete the function void update(int \*a,int \*b), which reads two integers as argument, and sets  with the sum of them, and  with the absolute difference of them.

**Input Format**

The input will contain two integers,  and , separated by a newline.

**Output Format**

You have to print the updated value of  and , on two different lines.

Note: Input/ouput will be automatically handled. You only have to complete the function described in the 'task' section.

**Sample Input**

4

5

**Sample Output**

9

1

# Solution:

#include <stdio.h>

void update(int \*a,int \*b) {

\*a=\*a+\*b;

\*b=abs(\*a-(2\*\*b));

}

int main() {

int a, b;

int \*pa = &a, \*pb = &b;

scanf("%d %d", &a, &b);

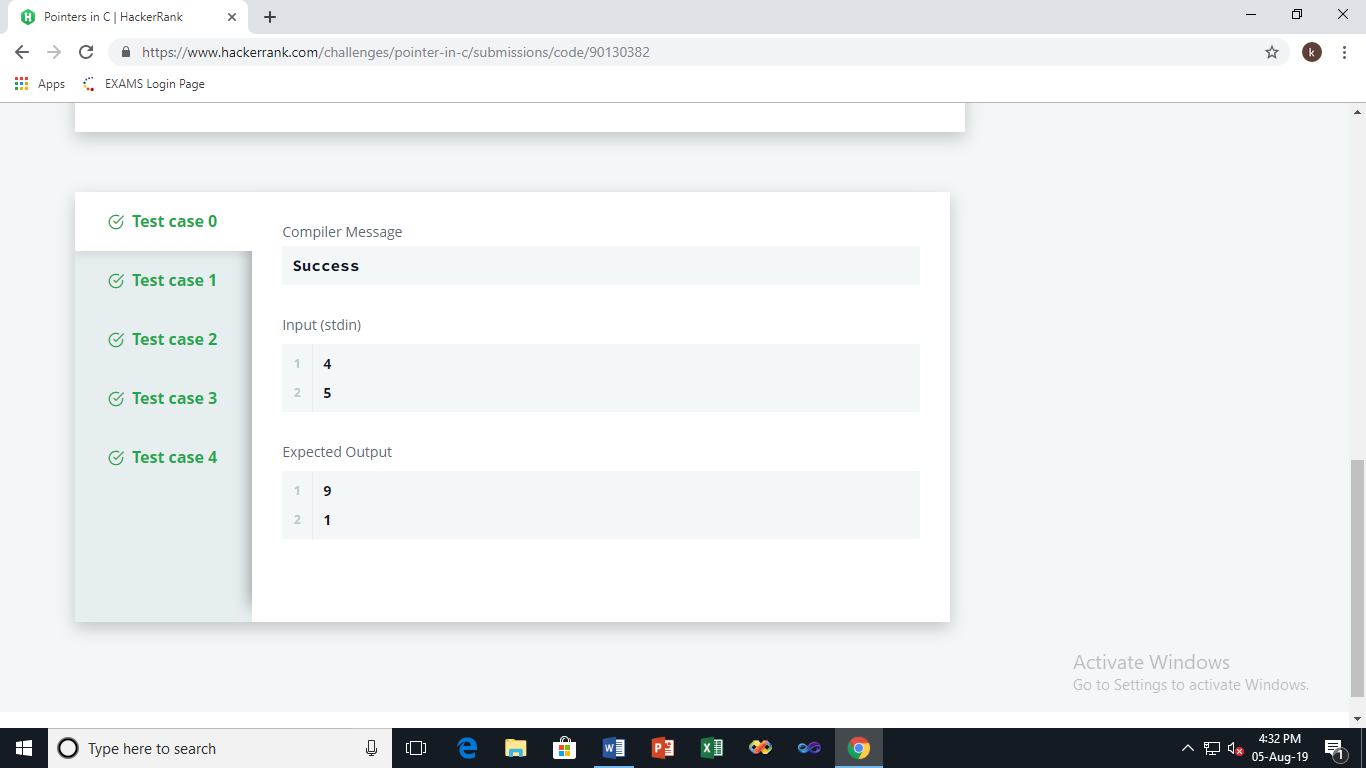
update(pa, pb);

printf("%d\n%d", a, b);

return 0;

}

Output:



Score: 10

# 6.For Loop in C

**Task**

For each integer  in the interval  (given as input) :

* If , then print the English representation of it in lowercase. That is "one" for , "two" for , and so on.
* Else if  and it is an even number, then print "even".
* Else if  and it is an odd number, then print "odd".

**Input Format**

The first line contains an integer, .   
The seond line contains an integer, .

**Constraints**

**Output Format**

Print the appropriate english representation,even, or odd, based on the conditions described in the 'task' section.

**Note:**

**Sample Input**

8

11

**Sample Output**

eight

nine

even

odd

# solution:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{

int a, b,i;

scanf("%d\n%d", &a, &b);

// Complete the code.

static const char\* strings[]={"one","two","three","four","five","six","seven","eight","nine"};

for(i=a;i<=b;i++)

{

if(i<=9){printf("%s\n",strings[i-1]);}

else if(i%2==0){printf("even\n");}

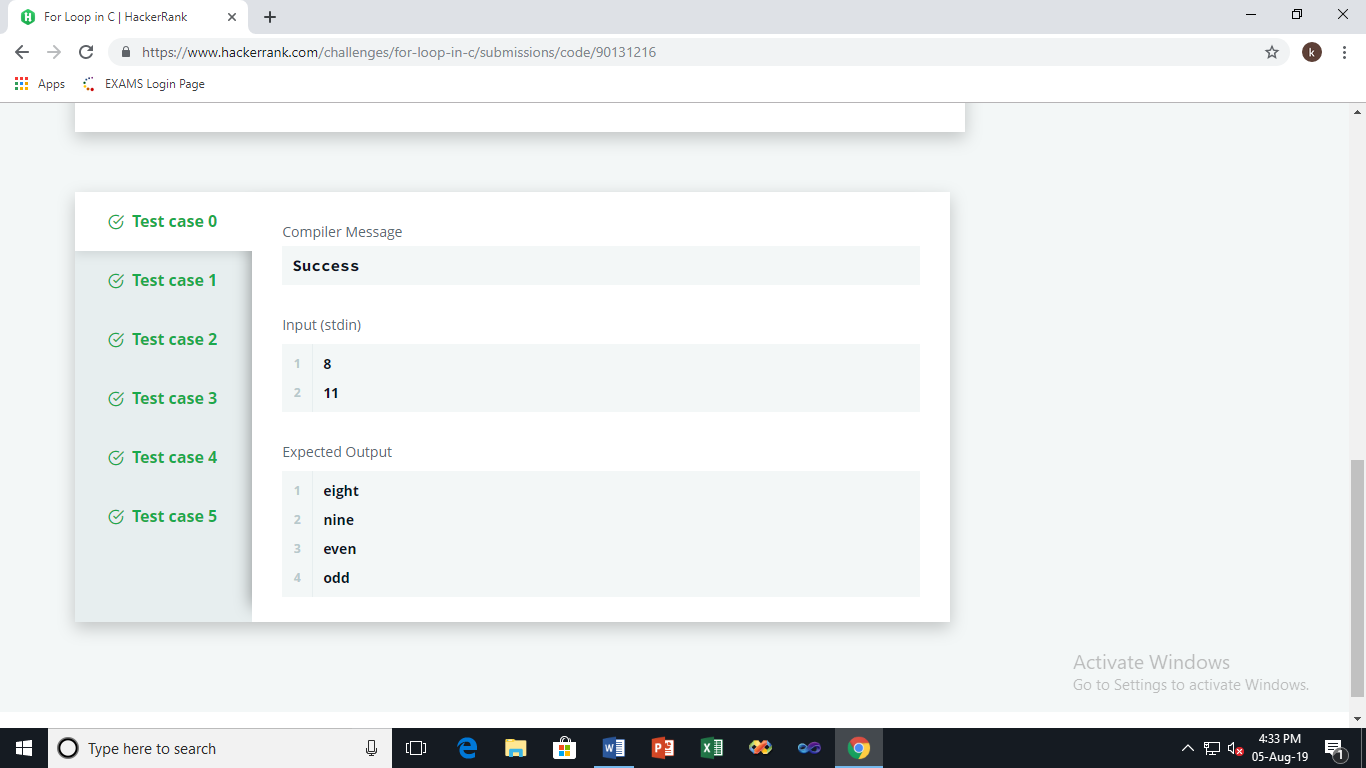
else{printf("odd\n");}

}

return 0;

}

Output:



Score: 10

# 7.Sum of Digits of a Five Digit Number

**Task**

In this challenge, you have to input a five digit number and print the sum of digits of the number.

**Input Format**

The input contains a single five digit number, .

**Constraints**

**Output Format**

Print the sum of the digits of the five digit number.

**Sample Input 0**

10564

**Sample Output 0**

16

# Solution:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main() {

int n;

scanf("%d", &n);

int digit,temp,sum=0;

temp=n;

while(temp>0)

{

digit=temp%10;

sum=sum+digit;

temp=temp/10;

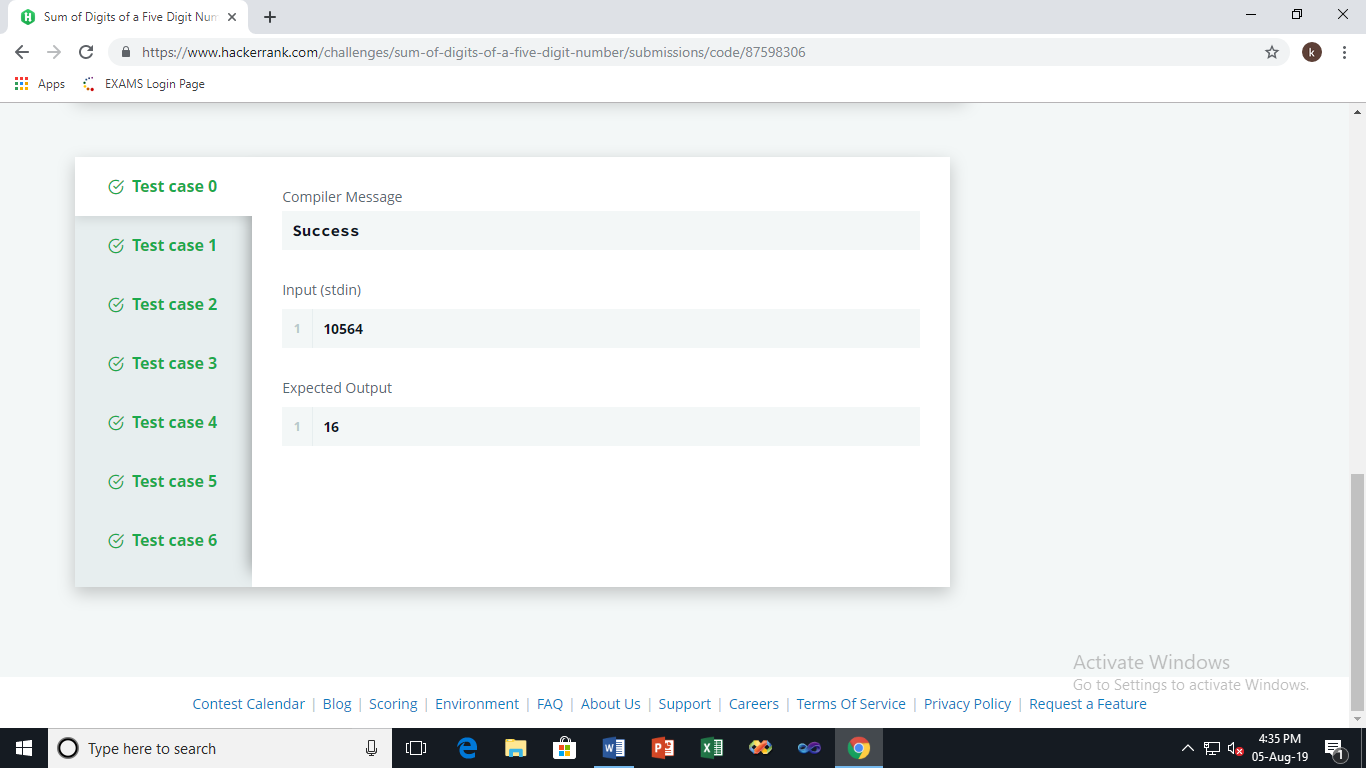
}

printf("%d\n",sum);

return 0;

}

Output:



Score: 15

# 8.Bitwise Operators

**Task**   
Given set , find:

* the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .
* the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .
* the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .

**Input Format**

The only line contains  space-separated integers,  and , respectively.

**Constraints**

**Output Format**

* The first line of output contains the maximum possible value of .
* The second line of output contains the maximum possible value of .
* The second line of output contains the maximum possible value of .

**Sample Input 0**

5 4

**Sample Output 0**

2

3

3

# SOLUTION:

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

//Complete the following function.

void calculate\_the\_maximum(int n, int k) {

int maxAnd = 0;

int maxOr = 0;

int maxXor = 0;

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

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

if (((i&j) > maxAnd) && ((i&j) < k)) {

maxAnd = i&j;

}

if (((i|j) > maxOr) && ((i|j) < k)) {

maxOr = i|j;

}

if (((i^j) > maxXor) && ((i^j) < k)) {

maxXor = i^j;

}

}

}

printf("%d\n%d\n%d\n", maxAnd, maxOr, maxXor);

//Write your code here.

}

int main() {

int n, k;

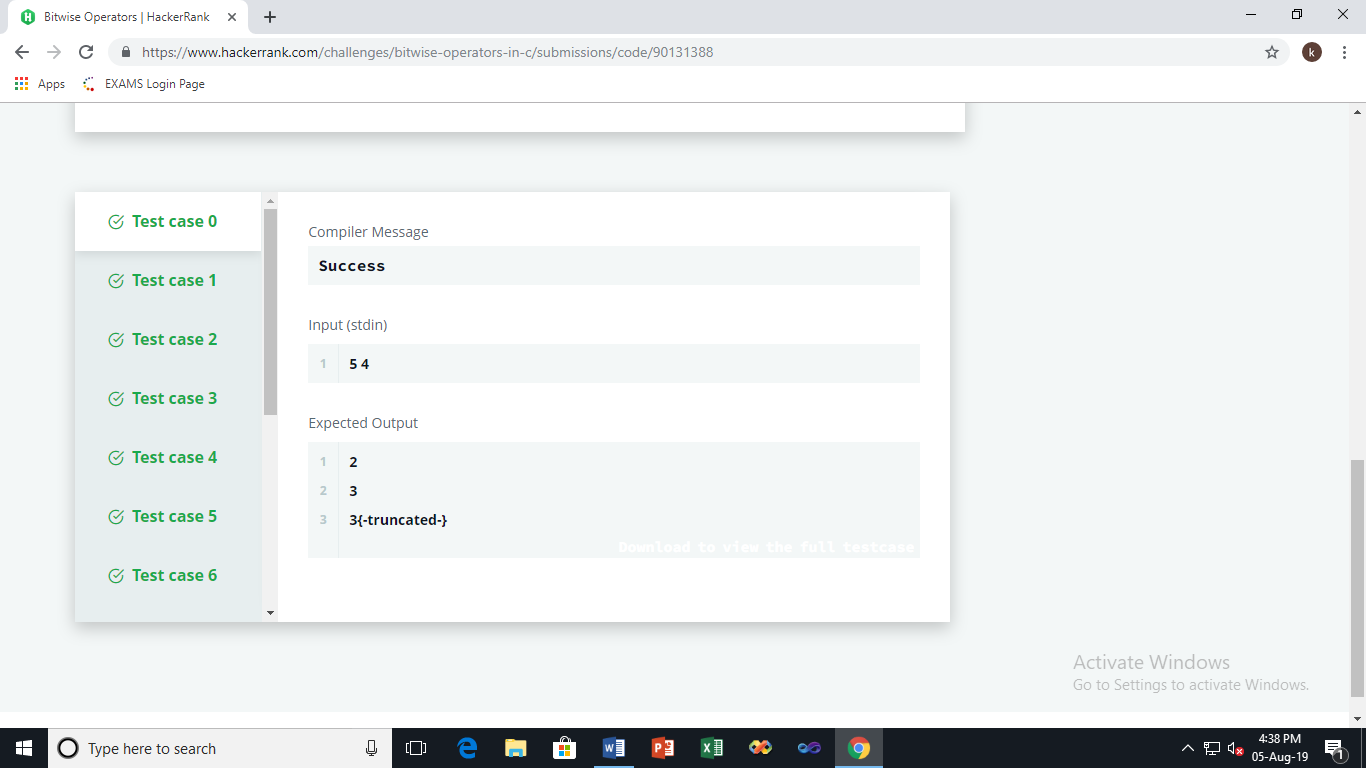
scanf("%d %d", &n, &k);

calculate\_the\_maximum(n, k);

return 0;

}

Output:



Score: 15

# 9.1D Arrays in C

**Input Format**

The first line contains an integer, .   
The next line contains  space-separated integers.

**Constraints**

**Output Format**

Print in a single line the sum of the integers in the array.

**Sample Input 0**

6

16 13 7 2 1 12

**Sample Output 0**

51

**Sample Input 1**

7

1 13 15 20 12 13 2

**Sample Output 1**

76

**SOLUTION:**

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main() {

/\* Enter your code here. Read input from STDIN. Print output to STDOUT \*/

int n,a[1000],i,sum=0;

scanf("%d",&n);

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

{

scanf("%d",&a[i]);

}

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

{

sum=sum+a[i];

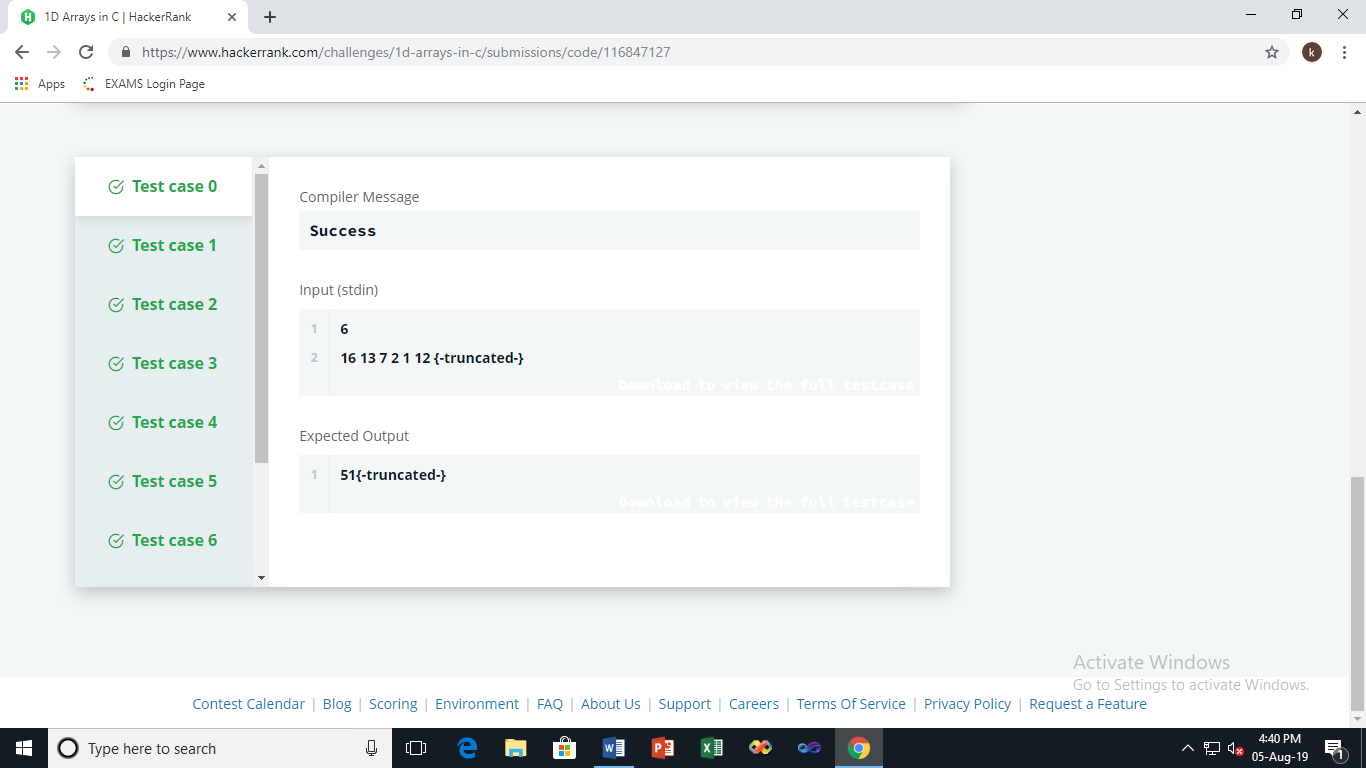
}

printf("%d",sum);

return 0;

}

Output:



Score:10

# 10. Students Marks Sum

**Input Format**

* The first line contains , denoting the number of students in the class, hence the number of elements in .
* Each of the  subsequent lines contains .
* The next line contains .

**Constraints**

* (where )
* =  or

**Output Format**

The output should contain the sum of all the aternate elements in  as explained above.

**Sample Input 0**

3

3

2

5

b

**Sample Output 0**

8

**Explanation 0**

 = [3, 2, 5] and  = .

So, .

**Sample Input 1**

5

1

2

3

4

5

g

**Sample Output 1**

6

**SOLUTION:**

int marks\_summation(int\* marks, int number\_of\_students, char gender) {

int sum=0;

if(gender=='b'){

for(int i=0;i< number\_of\_students;i++){

if(i%2==0){

sum+=marks[i];

}

}

} else if(gender=='g'){

for(int i=0;i<number\_of\_students;i++){

if(i%2>0){

sum+=marks[i];

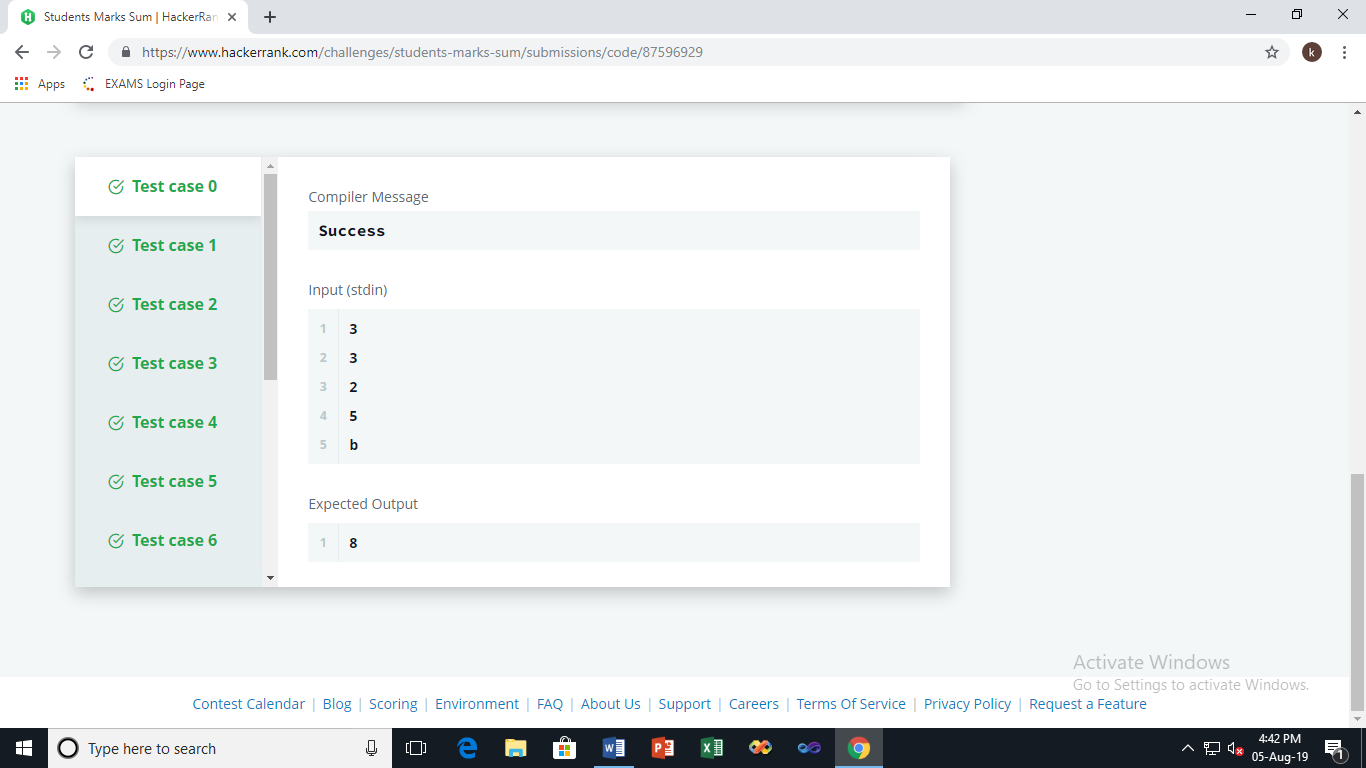
}

}

}return sum;

}

Output:



Score: 20