Question **3**

Correct

Marked out of 1.00

Flag question

To print unsigned values on the console, use %u format character instead of %d in the **printf()** function.

Whenever an attempt is made to assign a negative number to an **unsigned int** (For eg: unsigned int num = -1;) the compiler does not flag it as an **error**. Instead, it will automatically convert the negative number to a positive number as shown below:

```
unsigned int num = -1;
The value stored in num = unsigned
The final value in num = 4294967295
```

In the program given below, fill in the missing **format characters** to print **signed** and **unsigned** values.

Answer: (penalty regime: 0 %)

Reset answer

```
1  #include <stdio.h>
2
3  int main()
4  {
5      signed int number1 =
6      signed long long numb
7      printf("Given signed
8      printf("Given unsigne
9      return 0;
10 }
```

	Expected
✓	Given signed values are -20 Given unsigned values are 4

Passed all tests! ✓

Question **4**

Correct

Marked out of 1.00

Flag question

Identify the error and correct the code.
[Hint: Verify if all variables are declared before they are first used.**]**

Answer: (penalty regime: 0 %)

Reset answer

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int number1 = 20, num
6      sub = number1 - numbe
7      printf("The differenc
8      return 0;
9  }
10
```

	Expected
✓	The difference of the two g

Passed all tests! ✓

Finish review

Quiz navigation

Question **1**

Correct

Marked out of 1.00

Flag question

Identify and correct the errors in the code given below:

Expected Output:

Given float values are num1 = 5.340000, num2 = 125.789001
The result after dividing in float format = 23.555992
The result after dividing in exponential format = 2.355599e+01

Answer: (penalty regime: 0 %)

Reset answer

```
1  #include <stdio.h>
2
3  int main()
4  {
5      float num1 = 5.34, nu
6      printf("Given float v
7      result = num2 / num1;
8      printf("The result af
9      printf("The result af
10     return 0;
11 }
```

	Expected
✓	Given float values are num1 The result after dividing i The result after dividing i

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

Identify and correct the errors in the code given below:

Answer: (penalty regime: 0 %)

Reset answer

```
1  #include <stdio.h>
2
3  int main()
4  {
5      float num1 = 5.345f,
6      printf("Given float v
7      result = num1 / num2;
8      printf("Result of div
9      return 0;
10 }
```

	Expected
✓	Given float values are num1 Result of division = 0.4310

Passed all tests! ✓

Finish review

Quiz navigation

1 2

Question **1**

Correct

Marked out of 1.00

Flag question

Relational and equality operators are used to **test** or **compare** two numeric values or numeric expressions.

In **C, Relational and equality operators** when applied on the operands, produce an **integer** value which is either 0 or 1 and these are often referred to as logical values. The value 0 represents false and the value 1 represents true.

In **C**, there are **four** relational and **two** equality operators as given below:

Operator	Description
>	Checks for greater-than condition
>=	Checks for greater-than-or-equals condition
<	Checks for less-than condition
<=	Checks for less-than-or-equals condition.
==	Checks if two values are equal
!=	Checks if two values are unequal

The format for usage of **relational** and **equality operators** is as follows:

`operand1 operator operand2`

According to the coding conventions in C, a single space should be provided to the left and to the right of the operator.

The table given below demonstrates the use of various **relational and equality operators** using variables int num1 = 7;, float num2 = 5.5;, char ch = 'w':

Expression	Interpretation	Result Value
(num1 > 5)	true	1
((num1 + num2) <= 10)	false	0
(ch == 119)	true	1
(ch != 'p')	true	1
(ch >= 10 * (num1 + num2))	false	0

Read the code given below and retype in the space provided.

```
#include <stdio.h>

int main()
{
   int num1 = 7;
   float num2 = 5.5;
   char ch = 'w';
   printf("Result1 = %d\n", (num1 > 5));
   printf("Result2 = %d\n", ((num1 + num2) <= 10));
   printf("Result3 = %d\n", (ch == 119));
   printf("Result4 = %d\n", (ch != 'p'));
   printf("Result5 = %d", (ch >= 10 * (num1 + num2)));
   return 0;
}
```

Answer: (penalty regime: 0 %)

```
1  #include <stdio.h>
2  int main()
3  { int num1=7;
4  float num2=5.5;
5  char ch ='w';
6  Printf("Result1 = %d\n",(
7  Printf("Result2 = %d\n",(
8  Printf("Result3 = %d\n",(
9  Printf("Result4 = %d\n",(
10 Printf("Result5 = %d\n",(
11 return 0;
12 }
```

	Expected	Got
✓	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 1 Result5 = 0	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 1 Result5 = 0

Passed all tests! ✓

Finish review

Logical operators are used to perform logical operations on the given expressions.

An expression containing a logical operator returns
either 0 (or) 1 depending on the evaluation of the expression to either false or true respectively.

Note: In C, false is represented as 0 (zero) and all non-zero values can be treated as true.

Given below are the **three** logical operators in **C**:

Operator	Description	Meaning
&&	logical AND	It returns true when both conditions are true, else, it returns false
||	logical OR	It returns true if atleast one of the conditions is true
!	logical NOT	It returns true when the given expression is false and returns false when the given expression is true

According to the coding conventions in C, a single space should be provided to the left and to the right of the operator.

The below table demonstrates the use of various **relational and equality operators** using variables int num1 = 7;, float num2 = 5.5;, char ch = 'w':

Expression	Interpretation	Result Value
(num1 >= 6) && (ch == 'w')	true	1
(num2 < 11) && (num1 > 100)	false	0
(ch != 'p') || ((num1 + num2) <= 10)	true	1
!(num1 > (num2 + 1))	false	0
!(num1 <= 3)	true	1

Read the code given below and retype in the space provided.

```
#include <stdio.h>

int main()
{
   int num1 = 7;
   float num2 = 5.5;
   char ch = 'w';
   printf("Result1 = %d\n", ((num1 >= 6) && (ch == 'w')));
   printf("Result2 = %d\n", ((num2 < 11) && (num1 > 100)));
   printf("Result3 = %d\n", ((ch != 'p') || ((num1 + num2) <= 10)));
   printf("Result4 = %d\n", !(num1 > (num2 +1)));
   printf("Result5 = %d\n", !(num1 <= 3));
   return 0;
}
```

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main() {
    int n1 = 7;
    float n2 = 5.5;
    char ch = 'w';

    printf("Result1 = %d\
    printf("Result2 = %d\
    printf("Result3 = %d\
    printf("Result4 = %d\
    printf("Result5 = %d\

    return 0;
}
```

	Expected	Got	
✓	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 0 Result5 = 1	Result1 = 1 Result2 = 0 Result3 = 1 Result4 = 0 Result5 = 1	✓

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

Read the code given below to understand the working of unary operators. Retype in the space provided.

```
#include <stdio.h>

int main()
{
  int x = 16;
  printf("+x = %d\n", (+x));
  printf("-x = %d\n", (-x));
  printf("x = %d\n", x);
  printf("++x = %d\n", (++x));
  printf("x = %d\n", x);
  printf("x++ = %d\n", (x++));
  printf("x = %d\n", x);
  printf("--x = %d\n", (--x));
  printf("x = %d\n", x);
  printf("x-- = %d\n", (x--));
  printf("x = %d", x);
  return 0;
}
```

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
    int x= 16;
    printf("+x = %d\n",(+x
    printf("-x = %d\n",(-x
    printf("x = %d\n",x);
    printf("++x = %d\n",(+
    printf("x = %d\n",x);
    printf("x++ = %d\n",(x
    printf("x = %d\n",x);
    printf("--x = %d\n",(-
    printf("x = %d\n",x);
    printf("x-- = %d\n",(x
    printf("x = %d",x);
return 0;
}
```

	Expected	Got	
✓	+x = 16 -x = -16 x = 16 ++x = 17 x = 17 x++ = 17 x = 18 --x = 17 x = 17 x-- = 17 x = 16	+x = 16 -x = -16 x = 16 ++x = 17 x = 17 x++ = 17 x = 18 --x = 17 x = 17 x-- = 17 x = 16	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

Read the code given below to understand the working of **increment** and **decrement** operators. Retype in the space provided.

```
#include <stdio.h>

int main()
{
  int x = 4, y;
  y = x++;
  printf("y = %d x = %d\n", y, x);
  y = ++x;
  printf("y = %d x = %d\n", y, x);
  y = x--;
  printf("y = %d x = %d\n", y, x);
  y = --x;
  printf("y = %d x = %d\n", y, x);
  return 0;
}
```

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
    int x = 4,y;
    y=x++;
    printf("y = %d x = %d
    y= ++x;
    printf("y = %d x = %d
    y=x--;
    printf("y = %d x = %d
    y=--x;
    printf("y = %d x = %d
    return 0;
}
```

	Expected	Got	
✓	y = 4 x = 5 y = 6 x = 6 y = 6 x = 5 y = 4 x = 4	y = 4 x = 5 y = 6 x = 6 y = 6 x = 5 y = 4 x = 4	✓

Passed all tests! ✓

Finish review

Quiz navigation

1 2

Question 1

Correct

Marked out of 1.00

Flag question

Read the code given below to understand the usage of the assignment operator. Retype in the space provided.

```
#include <stdio.h>

int main()
{
  int x = 24, y = 39, z = 45;
  z = x + y;
  y = z - y;
  x = z - y;
  printf("x = %d y = %d z = %d", x, y, z);
  return 0;
}
```

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main() {
    int x = 24, y = 39, z

    z = x + y;
    y = z - y;
    x = z - y;

    printf("x = %d y = %d

    return 0;
}
```

	Expected	Got
✓	x = 39 y = 24 z = 63	x = 3

Passed all tests! ✓

Question 2

Correct

Marked out of 1.00

Flag question

Read the code given below and retype in the space provided.

```
#include <stdio.h>

int main()
{
  int x = 2, y = 18, z = 12;
  x += y;
  printf("x = %d\n", x);
  y *= 2;
  printf("y = %d\n", y);
  z /= 5;
  printf("z = %d\n", z);
  x %= 7;
  printf("x = %d", x);
  return 0;
}
```

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main() {
    int x = 2, y = 18, z

    x += y;
    printf("x = %d\n", x)

    y *= 2;
    printf("y = %d\n", y)

    z /= 5;
    printf("z = %d\n", z)

    x %= 7;
    printf("x = %d\n", x)

    return 0;
}
```

	Expected	Got	
✓	x = 20 y = 36 z = 2 x = 6	x = 20 y = 36 z = 2 x = 6	✓

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

There was a large ground in center of the city which is rectangular in shape. The Corporation decides to build a Cricket stadium in the area for school and college students, But the area was used as a car parking zone. In order to protect the land from using as an unauthorized parking zone, the corporation wanted to protect the stadium by building a fence. In order to help the workers to build a fence, they planned to place a thick rope around the ground. They wanted to buy only the exact length of the rope that is needed. They also wanted to cover the entire ground with a carpet during rainy season. They wanted to buy only the exact quantity of carpet that is needed. They requested your help. Can you please help them by writing a program to find the exact length of the rope and the exact quantity of carpet that is required?

Input format:

Input consists of 2 integers. The first integer corresponds to the length of the ground and the second integer corresponds to the breadth of the ground.

Output Format:

Output Consists of two integers. The first integer corresponds to the length. The second integer corresponds to the quantity of carpet required.

Sample Input:

50

20

Sample Output:

140

1000

For example:

Input	Result
50 20	140 1000

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{int a,b,c,d;
scanf("%d",&a);
scanf("%d",&b);
c=(2*(a+b));
d=a*b;
printf("%d\n%d",c,d);
}
```

	Input	Expected	Got	
✓	50 20	140 1000	140 1000	✓

Passed all tests! ✓

Finish review

Quiz navigation

Training for sports day has begun and the physical education teacher has decided to conduct some team games. The teacher wants to split the students in higher secondary into equal sized teams. In some cases, there may be some students who are left out from the teams and he wanted to use the left out students to assist him in conducting the team games. For instance, if there are 50 students in a class and if the class has to be divided into 7 equal sized teams, 7 students will be there in each team and 1 student will be left out. That 1 student will assist the PET. With this idea in mind, the PET wants your help to automate this team splitting task. Can you please help him out?

INPUT FORMAT:

Input consists of 2 integers. The first integer corresponds to the number of students in the class and the second integer corresponds to the number of teams.

OUTPUT FORMAT:

The output consists of two integers. The first integer corresponds to the number of students in each team and the second integer corresponds to the students who are left out.

SAMPLE INPUT:

60

8

SAMPLE OUTPUT:

7

4

For example:

Input	Result
60 8	7 4

Answer: (penalty regime: 0 %)

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

	Input	Expected	Got	
✓	60 8	7 4	7 4	✓

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

Each Sunday, a newspaper agency sells w copies of a special edition newspaper for Rs.x per copy. The cost to the agency of each newspaper is Rs.y. The agency pays a fixed cost for storage, delivery and so on of Rs.100 per Sunday. The newspaper agency wants to calculate the profit which it obtains only on Sundays. Can you please help them out by writing a program to compute the profit if w, x, and y are given?

INPUT FORMAT:

Input consists of 3 integers: w, x, and y. w is the number of copies sold, x is the cost per copy and y is the cost the agency spends per copy.

OUTPUT FORMAT:

The output consists of a single integer which corresponds to the profit obtained by the newspaper agency.

SAMPLE INPUT:

1000

2

1

SAMPLE OUTPUT:

900

For example:

Input	Result
1000 2 1	900

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{int w,x,y,a,b;
scanf("%d%d%d",&w,&x,&y);
a=(w*y)+100;
b=(w*x)-a;
printf("%d",b);
}
```

	Input	Expected	Got	
✓	1000 2 1	900	900	✓

Passed all tests! ✓

Finish review

uiz navigation

1

Finish review

Question 1

Correct

Marked out of 1.00

Flag question

Four kids Peter, Susan, Edmond and Lucy travel through a wardrobe to the land of Narnia. Narnia is a fantasy world of magic with mythical beasts and talking animals.While exploring the land of narnia Lucy found Mr.Tumnus the two legged stag ,and she followed it, down a narrow path .She and Mr.Tumnus became friends and he offered a cup of coffee to Lucy in his small hut.It was time for Lucy to return to her family and so she bid good bye to Mr.Tumnus and while leaving Mr.Tumnus told that it is quite difficult to find the route back as it was already dark. He told her to see the trees while returning back and said that the first tree with two digits number will help her find the way and the way to go back to her home is the sum of digits of the tree and that numbered way will lead her to the tree next to the wardrobe where she can find the others. Lucy was already confused, so please help her in finding the route to her home....

Input Format:

Input consists of an integer corresponding to the 2-digit number.

Output Format:

Output consists of an integer corresponding to the sum of its digits.

SAMPLE INPUT:

87

SAMPLE OUTPUT:

15

For example:

Input	Result
87	15

Answer: (penalty regime: 0 %)

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

	Input	Expected	Got	
✓	87	15	15	✓
✓	54	9	9	✓

Passed all tests! ✓

Finish review

uiz navigation

Question **1**

Correct

Marked out of 1.00

Flag question

In the program given below, we shall learn how to assign values to int data type from binary, octal, hex and character literals.

Read the code given below and retype in the space provided.

```
#include <stdio.h>
int main()
{
   int binaryThree = 0b11;
   printf("binaryThree value = %d\n", binaryThree);
   int octalEight = 010;
   printf("octalEight value = %d\n", octalEight);
   int hexTen = 0xA;
   printf("hexTen value = %d\n", hexTen);
   int asciiValueOfOne = '1';
   printf("asciiValueOfOne value = %d\n", asciiValueOfOne);
   int asciiValueOfA = 'A';
   printf("asciiValueOfA value = %d\n", asciiValueOfA);
   return 0;
}
```

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  int main()
3  { int binaryThree = 0b11;
4  printf("binaryThree value
5  int octalEight = 010;
6  printf("octalEight value
7  int hexTen = 0xA;
8  printf("hexTen value = %d
9  int asciiValueOfOne='1';
10 printf("asciiValueOfOne v
11 int asciiValueOfA = 'A';
12 printf("asciiValueOfA val
13 return 0;
14 }
```

	Expected
✓	binaryThree value = 3 octalEight value = 8 hexTen value = 10 asciiValueOfOne value = 49 asciiValueOfA value = 65

Passed all tests! ✓

Question **2**

Correct

Marked out of 1.00

Flag question

In the program given below, fill in the missing code to add two integer numbers.

Answer: (penalty regime: 0 %)

Reset answer

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int num1 = 15, num2 =
6      printf("Given integer
7      //Write the code to a
8      sum=num1+num2;
9      printf("Sum of 2 give
10     return 0;
11 }
```

	Expected
✓	Given integers are num1 = 1 Sum of 2 given numbers = 40

Passed all tests! ✓

Finish review

Question **1**

Correct

Marked out of 1.00

Flag question

Each Sunday, a newspaper agency sells X copies of a certain newspaper for Rs.A per copy. The cost to the agency of each newspaper is Rs.B . The agency pays a fixed cost for storage, delivery and so on of Rs.100 per Sunday. The newspaper agency wants to calculate the profit obtained on Sundays. Can you please help them out by writing a C program to compute the profit given X, A and B.

Input Format:

Input consists of 3 integers: X, A and B. X is the number of copies sold, A is the cost per copy and B is the cost the agency spends per copy.

Output Format:

Refer Sample Input and Output for exact formatting specifications.

Sample Input and Output:

Input

1000

2

1

Output

900

For example:

Input	Result
1000 2 1	900

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{int x,a,b;
scanf("%d %d %d",&x,&a,&b
int revenue= x*a;
int cost=(x*b)+100;
int profit=revenue-cost;
printf("%d",profit);
}
```

	Input	Expected	Got	
✓	1000 2 1	900	900	✓

Passed all tests! ✓

Question **1**

Correct

Marked out of 1.00

Flag question

Baba is very kind to beggars and every day Baba donates half of the amount he has when ever a beggar requests him. The money M left in Baba's hand is passed as the input and the number of beggars B who received the alms are passed as the input. The program must print the money Baba had in the beginning of the day.

Input Format:

The first line denotes the value of M.
The second line denotes the value of B.

Output Format:

The first line denotes the value of money with Baba in the beginning of the day.

Example Input/Output:

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>
#include <math.h>
int main ()
{int m,b;
scanf("%d %d",&m,&b);
int initialmoney=m*pow(2 ,
printf("%d",initialmoney);
}
```

	Input	Expected	Got	
✓	100 2	400	400	✓

Passed all tests! ✓

Question **1**

Incorrect

Marked out of 1.00

Flag question

The CEO of company ABC Inc wanted to encourage the employees coming on time to the office. So he announced that for every consecutive day an employee comes on time in a week (starting from Monday to Saturday), he will be awarded Rs.200 more than the previous day as "Punctuality Incentive". The incentive I for the starting day (ie on Monday) is passed as the input to the program. The number of days N an employee came on time consecutively starting from Monday is also passed as the input. The program must calculate and print the "Punctuality Incentive" P of the employee.

Input Format:

The first line denotes the value of I.
The second line denotes the value of N.

Output Format:

The first line denotes the value of P.

Example Input/Output:

Input:

500
3

Output:

2100

Explanation:

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

So total = Rs.2100

For example:

Input	Result
500 3	2100
100 3	900

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{int I,P,N,incentive;
scanf("%d",&I);
scanf("%d",&N);
P=I;
incentive=0;
for(int I=0;I<N;I++) {
incentive += P;
P+=200;}
printf("%d\n",incentive);
return 0;
}
```

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

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.

Question **1**

Correct

Marked out of 1.00

Flag question

Bajan Lal distributes C chocolates to school N students every Friday. The C chocolates are distributed among N students equally and the remaining chocolates R are given back to Bajan Lal.

As an example if C=100 and N=40, each student receives 2 chocolates and the balance 100-40*2 = 20 is given back.

If C=205 and N=20, then each student receives 10 chocolates and the balance 205-20*10 = 5 is given back.

Help the school to calculate the chocolates to be given back when C and N are passed as input.

Input Format:

The first line denotes C
The second line denotes N

Output Format:

The first line denotes R - the number of chocolates to be given back.

Example Input/Output:

Input:

300
45

Output:

30

Answer: (penalty regime: 0 %)

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

	Input	Expected	Got	
✓	300 45	30	30	✓

Passed all tests! ✓

Question **1**

Incorrect

Marked out of 1.00

Flag question

The general format of if statement is

```
if (condition) {
   statement-1;
   statement-2;
   ....
   statement-n;
}
```

The if construct is a **selective statement**, the statements within the block are executed only once when the **condition evaluates to true**, otherwise the control goes to the first statement after the if construct.

If only one statement is presented in the if construct then there is no need to specify the braces {, } i.e., if braces are not specified for the if construct, by default the next immediate statement is the only statement considered for the if construct.

Below code prints the number only when it is **divisible by 3**:

```
#include <stdio.h>
int main()
{
        int num;
        printf("Enter a number : ")
        scanf("%d", &num);
        if (num % 3 == 0)
        {

                printf("Given numbe
        }
        return 0;
}
```

In the above code, num % 3 == 0 is the **condition**, which verifies whether the **number is divisible by 3**. Only if the condition returns 1 (true) then the control enters in to the **if-block** and executes the statement.

Fill in the missing code in the below program to check whether the given number is divisible by **3** or not.

For example:

Input	Result
9	Given number 9 is divisible
7	Given number 7 is not divisi

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int num,num1;
    scanf("%d %d ", &num,
   if (num%3==0)
    {
        printf("Given num
    }
    {
        printf("Given num
    }
    return 0;
}
```

	Input	Expected
✗	9	Given number 9 is div
✗	7	Given number 7 is not

Some hidden test cases failed, too.
Your code must pass all tests to earn any marks. Try again.

Show differences

Finish review

Question 1

Correct

Marked out of 1.00

Flag question

A switch statement is used to change the control flow of a program execution through multiple paths depending on an expression's value.

The below code demonstrates how to use a switch-case construct to print the corresponding English words for the digits (1 to 9) read from the standard input.

One way is to write a long nested if-else-if for the 10 numbers or the other way is to use a switch-case statement.

See and retype the below code which demonstrates the usage of switch statement to print the English word of the given number between 1 to 9.

```
#include <stdio.h>
int main()
{
  int value;
  scanf("%d", &value);
  switch (value)
  {
  case 1:
    printf("One");
    break;
  case 2:
    printf("Two");
    break;
    case 3:
    printf("Three");
    break;
    case 4:
    printf("Four");
      break;
    case 5:
    printf("Five");
    break;
    case 6:
    printf("Six");
    break;
    case 7:
    printf("Seven");
      break;
    case 8:
    printf("Eight");
      break;
    case 9:
      printf("Nine");
      break;
    case 10:
      printf("Ten");
      break;
    default:
      printf("Number %d is not in the
range 1 to 10", value);
  }
  return 0;
}
```

For example:

Input	Result
2	Two
9	Nine
15	Number 15 is not in the rang

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{int value;
scanf("%d",&value);
switch(value)
{case 1:
printf("One");
break;
case 2:
printf("Two");
break;
case 3:
printf("Three");
break;
case 4:
printf("Four");
break;
case 5:
printf("Five");
break;
case 6:
printf("Six");
break;
case 7:
printf("Seven");
break;
case 8:
printf("Eight");
break;
case 9:
printf("Nine");
break;
case 10:
printf("Ten");
break;
default: printf("Number %
}
}
```

	Input	Expected
✓	2	Two
✓	9	Nine
✓	15	Number 15 is not in t

Passed all tests! ✓

Question 2

Correct

Marked out of 1.00

Flag question

Assume that the weekdays are provided with the below numbers:

```
Sunday — 0
Monday — 1
Tuesday — 2
Wednesday — 3
Thursday — 4
Friday — 5
Saturday — 6
```

Write a program to read the **weekday number** from the standard input and print the **weekday name** using switch-case.

For example, if the user gives the input as 1:

```
1
```

then the program should print the result as:

```
Monday
```

Note: if the given input number is not in the range i.e., other than 0 to 6, the output should be as given below:

```
Invalid weekday number
```

For example:

Input	Result
6	Saturday
0	Sunday
7	Invalid weekday number

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main()
{int days;
scanf("%d",&days);
switch(days)
{ case 0:
printf("Sunday");
break;
case 1:
printf("Monday");
break;
case 2:
printf("Tuesday");
break;
case 3:
printf("Wednesday");
break;
case 4:
printf("Thursday");
break;
case 5:
printf("Friday");
break;
case 6:
printf("Saturday");
break;
default:
printf("Invalid weekday n
}
}
```

	Input	Expected
✓	6	Saturday
✓	0	Sunday
✓	7	Invalid weekday numbe

Passed all tests! ✓

Finish review

Question 1

Correct

Marked out of 1.00

Flag question

Most of the programming languages provide a special construct/statement using which we can repeatedly execute one or more statement as long as a condition is **true**. In C, we have while, do-while and for as the three main looping constructs or statements.

Below is a general syntax for using a while statement:

```
while (condition)
{
        statement_1;
        statement_2;
        ....
}
```

The block of code inside the opening and closing brace which follows the while-statement is called the **while-loop** body.

A while statement is used to execute some code repeatedly as long as a condition evaluates to true.

The condition is an expression which should always evaluate to either true or false.

- If it evaluates to true, the body containing one or more code statements is executed.
- If the expression evaluates to false, the control skips executing the **while-loop** body.

The while-loop construct is also referred to as an entry controlled loop. Meaning, first the condition is evaluated and only if the condition evaluates to true the body of the loop is executed. After executing the body the control is automatically transferred back to the condition and the process continues until the condition evaluates to false.

See and retype the below code which uses a while-loop to read multiple numbers from standard input and prints their sum when the **sum** exceeds 100.

```
#include <stdio.h>

int main()
{
   int total = 0;
   while (total <= 100)
   {
   int num;
   scanf("%d", &num);
      total += num;
   }
   printf("The total of given numbers is :
%d", total);
   return 0;
}
```

For example:

Input	Result
34 62 24	The total of given numbers i

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
int main()
{ int total=0;
while (total<=100)
{ int num;
scanf("%d",&num);
total+=num;
}
printf("The total of give
}
```

	Input	Expected
✓	34 62 24	The total of given nu

Passed all tests! ✓

Question 2

Correct

Marked out of 1.00

Flag question

The below sample code should print Ganga by number of times, where as the input is read by the programmer using **scanf()**.

Fill in the missing code so that it produces the desired output.

For example:

Input	Result
3	Ganga Ganga Ganga

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>
int main()
{
    int i = 0, n;
    scanf(  "%d",&n ); //
    while ( i<n  )
    { // complete the con
        printf("Ganga\n")
        i++; // Complete
    }
    return 0;
}
```

	Input	Expected	Got	
✓	3	Ganga Ganga Ganga	Ganga Ganga Ganga	✓

Passed all tests! ✓

Question **3**

Correct

Marked out of 1.00

Flag question

Write a **C** program to print first n **natural numbers**.

For example, if the user gives the **input** as :

```
3
```

then the program should **print** the result as:

```
The natural numbers from 1 - 3 : 1
```

For example:

Input	Result
3	The natural numbers from 1 -
9	The natural numbers from 1 -

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
int i=1,n;
scanf("%d",&n);
printf("The natural numbe
while(i<=n)
{printf("%d ",i);
i++;
}
printf("\n");
}
```

	Input	Expected
✓	3	The natural numbers f
✓	9	The natural numbers f

Passed all tests! ✓

Question **4**

Correct

Marked out of 1.00

Flag question

The below sample code should find the sum of **even numbers** between any two numbers.

[**Hint:** The numbers should be read by using scanf()].

Fill in the missing code so that it produces the desired output.

For example:

Input	Result
3 6	The sum of even integers bet

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int num1, num2, sum =
    scanf("%d %d",&num1,&

    while (num1<=num2  )
    { // Write the condit
 if (num1%2==0){
 sum = sum +num1 ;
        num1 += 2 ;
        continue;
    }
    num1+=1;
    }
    printf("The sum of ev
    return 0;
}
```

	Input	Expected
✓	3 6	The sum of even integ

Passed all tests! ✓

Question **5**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below program to read an **integer number** and find the reverse of the given number.

For example if the input is 1234, then the output will be 4321.

Hints

The logic of reversing of any number is pretty simple if you know how to find last digit of any number. Initially the variable reverse contains zero(0), the process of reversing involves four basic steps:

- Multiply the reverse variable by 10.
- Find the last digit of the given number by applying % 10.
- Add the last digit just found to reverse.
- Divide the original number by 10 to eliminate the last digit, which is not needed anymore.

Repeat the above four steps till the original number becomes 0 and finally we will be left with the reversed number in reverse variable.

For example:

Input	Result
1234	The reverse number of a give
765	The reverse number of a give

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int n, digit, reverse
    scanf("%d", &n);
    while ( n!=0 )
    { // Write the condit
    digit = n%10 ; // Fil
    reverse = reverse*10+
    n =n/10  ; // Fill th
    }
    printf("The reverse n
    return 0;
}
```

	Input	Expected
✓	1234	The reverse number of
✓	765	The reverse number of

Passed all tests! ✓

Question **6**

Correct

Marked out of 1.00

Flag question

Fill in the missing code in the below sample program which finds the factorial of a given number.

Factorial of a non-negative integer n, denoted by n!, is the product of all positive integers less than or equal to n. For example, 5! = 5 * 4 * 3 * 2 * 1 = 120.

The below sample code computes the factorial of a given non-zero integer.

The main() function declares an integer variable factorial and initializes it to 1, which it will use to store the computed factorial value.

It uses a **while-loop** to iterate from 2 to n multiplying the loop counter in each iteration with the factorial and storing the product again in factorial.

For example:

Input	Result
2	Factorial of given number 2
4	Factorial of given number 4

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int i, n, factorial =
    scanf("%d", &n);
    i = 2;
    while (i<=n )
    { // Write the condit
    factorial *=i  ; // F
    i++;
    }
    printf("Factorial of
    return 0;
}
```

	Input	Expected
✓	2	Factorial of given nu
✓	4	Factorial of given nu

Passed all tests! ✓

Question **7**

Correct

Marked out of 1.00

Flag question

Below partial code is to verify if the given number is a prime number or not.

A prime number is a positive integer greater than 1, which is not divisible by any other number other than 1 and itself. Examples of a few prime numbers are 2, 3, 5, 7, 11, 13, 17, 19, etc.

Fill in the missing code so that it produces the desired output.

For example:

Input	Result
7	The given number 7 is a prim
119	The given number 119 is not

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int n, i =2, count =0
    scanf("%d", &n);
    while (i<n)
    { // complete the con
    if (n%i==0 )
        { // complete the
        count++;
    }
    i++;
    }
    if (count == 0  )
    { // complete the con
        printf("The given
    }
    else
    {
    printf("The given num
    }
    return 0;
}
```

	Input	Expected
✓	7	The given number 7 is
✓	119	The given number 119

Passed all tests! ✓

Question **8**

Correct

Marked out of 1.00

Flag question

Below partial code is to verify if the given number is an armstrong number or not.

An armstrong number is a number that is the sum of its own digits raised to the power of number of digits that make up the original number.

For example, if the given number is 153, the total number of digits are 3, and the sum of cubes of each digit ($1^3 + 5^3 + 3^3$) is equal to the same number 153. Such a number is known as an armstrong number.

Let us take another example, if the given number is 9474, the total number of digits are 4, and the sum of the power of 4 of each digit ($9^4 + 4^4 + 7^4 + 4^4$) is equal to the same number 9474. Such a number is known as an armstrong number.

Similarly,
$9 = 9^1 = 9$
$371 = 3^3 + 7^3 + 1^3 = 27 + 343 + 1 = 371$
$3\ 8208 = 8^4 + 2^4 + 0^4 + 8^4 = 4096 + 16 + 0 + 4096 = 8208$

Fill in the missing code so that it produces the desired output.

For example:

Input	Result
777	The given number 777 is not
9	The given number 9 is an arm

Answer: (penalty regime: 0 %)

Reset answer

```
#include <stdio.h>

int main()
{
    int number, temp, rem
    scanf("%d", &number);
    temp = number;
    while ( temp!=0 )
    { // complete the con
    digits+=1 ; // increm
    temp = temp/10; //cal
    }
    temp = number;
    while (temp!=0 )
    { // complete the con
    rem = temp % 10;
    i = 1;
    power = 1;
    while (i<=digits )
        { // find the pow
        power = power*rem
            i++ ; // incr
        }
        sum +=power ; //
        temp=temp/10 ; //
    }
    if (sum== number )
    { // write the condit
        printf("The given
    }
    else
    {
    printf("The given num
    }
    return 0;
}
```

	Input	Expected
✓	777	The given number 777
✓	9	The given number 9 is

Passed all tests! ✓

Finish review

Quiz navigation

1 2 3 4 5 6 7 8