Ex:1 **Date**: **1.08.24**

Name: Veronica Regina Paul **Register number**:230701377

BASIC PROGRAMMING

1.

Aim:
To write a program that swaps two numbers using a temporary variable.

Algorithm:

1. Read two integers, a and b.
2. Store a in a temporary variable, temp.
3. Assign the value of b to a.
4. Assign the value of temp to b.
5. Print the swapped values of a and b.

Problem

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

For example:

Input	Result
10 20	20 10

PROGRAM

Answer: (penalty regime: 0 %)

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

}
```

OUTPUT

	Input	Expected	Got	
✔	10 20	20 10	20 10	✔

2.

Aim:
To determine if a candidate is eligible based on their marks in three subjects and total marks.

Algorithm:

1. Read marks of three subjects: maths, physics, and chemistry.
2. Calculate the total marks.
3. Check if:
    - maths >= 65, physics >= 55, and chemistry >= 50, OR
    - total >= 180.
4. If either condition is true, print "The candidate is eligible." Otherwise, print "The candidate is not eligible."

Problem

Write a C program to find the eligibility of admission for a professional course based on the following criteria:

Marks in Maths >= 65
Marks in Physics >= 55
Marks in Chemistry >= 50
Or
Total in all three subjects >= 180

Sample Test Cases

Test Case 1

Input

70 60 80

Output

The candidate is eligible

Test Case 2

Input

50 80 80

Output

The candidate is eligible

Test Case 3

Input

50 60 40

Output

The candidate is not eligible

PROGRAM

```c
#include<stdio.h>
int main(){
    int Math;
    int Physics;
    int Chemistry;
    int Total;
    scanf("%d",&Math);
    scanf("%d",&Physics);
    scanf("%d",&Chemistry);
    Total=Math+Physics+Chemistry;
    if((Math>=65 && Physics>=55 && Chemistry>=50)||(Total>=180)){
        printf("The candidate is eligible");

    }
    else{
        printf("The candidate is not eligible");
    }
}
```

OUTPUT

	Input	Expected	Got	
✔	70 60 80	The candidate is eligible	The candidate is eligible	✔
✔	50 80 80	The candidate is eligible	The candidate is eligible	✔

3.

Aim:

To calculate the discount for a purchase amount greater than 2000.

Algorithm:

1. Read the purchase amount B.
2. If B > 2000, calculate a 10% discount and print the discounted price.
3. Otherwise, print the original price.

PROBLEM

Malini goes to BestSave hyper market to buy grocery items. BestSave hyper market provides 10% discount on the bill amount B when ever the bill amount B is more than Rs.2000.

The bill amount B is passed as the input to the program. The program must print the final amount A payable by Malini.

Input Format:

The first line denotes the value of B.

Output Format:

The first line contains the value of the final payable amount A.

Example Input/Output 1:

Input:

1900

Output:

1900

Example Input/Output 2:

Input:

3000

Output:

2700

PROGRAM

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

OUTPUT

	Input	Expected	Got	
✔	1900	1900	1900	✔
✔	3000	2700	2700	✔

4.

Aim:

To calculate the growth of a value M by doubling it B times.

Algorithm:

1. Read integers M and B.
2. Initialize i to 0.
3. Repeat until i < B:
    - Multiply M by 2.
    - Increment i.
4. Print the final value of M.

Problem

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.

PROGAM

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

OUTPUT

	Input	Expected	Got	
✔	100 2	400	400	✔

5.

Aim:

To compute the sum of an incrementally increasing series.

Algorithm:

1. Read integers I and N.
2. Initialize t = 0 and a = 0.
3. Repeat while a < N:
    - Add I to t.
    - Increment I by 200.
    - Increment a.
4. Print the total sum t.

Problem

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

PROGRAM

```c
#include<stdio.h>
int main(){
    int a,b;
    scanf("%d%d",&a,&b);
    printf("%d",(a*b)+(b*200));

}
```

OUTPUT

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

6.

Aim:
To find multiples of a number in reverse order within a range.

Algorithm:

1. Read integers a, b, and c.
2. Loop from b to a in reverse order.
3. If a number is divisible by c, print it.

PROBLEM

Two numbers M and N are passed as the input. A number X is also passed as the input. The program must print the numbers divisible by X from N to M (inclusive of M and N).

Input Format:

The first line denotes the value of M
The second line denotes the value of N
The third line denotes the value of X

Output Format:

Numbers divisible by X from N to M, with each number separated by a space.

Boundary Conditions:

1 <= M <= 9999999
M < N <= 9999999
1 <= X <= 9999

Example Input/Output 1:

Input:
2
40
7

Output:
35 28 21 14 7

Example Input/Output 2:

Input:
66
121
11

Output:
121 110 99 88 77 66

PROGRAM

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

OUTPUT

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

7.

Aim:
To calculate the quotient and remainder of two numbers.

Algorithm:

1. Read two integers num1 and num2.
2. Calculate q = num1 / num2.
3. Calculate r = num1 % num2.
4. Print q and r.

Problem:

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

For example:

Input	Result
12 3	4 0

PROGRAM

```
#include<stdio.h>
int main()
{
    int num1;
    int num2;
    scanf("%d",&num1);
    scanf("%d",&num2);
    int q = num1 / num2;
    printf("%d \n",q);
    int r = num1 % num2;
    printf("%d",r);
}
```

OUTPUT:

	Input	Expected	Got	
✔	12 3	4 0	4 0	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

8.

Aim:
To find the largest number among three integers.

Algorithm:

1. Read three integers num1, num2, and num3.
2. Compare the numbers:
    - If num1 > num2 and num1 > num3, print num1.
    - Else if num2 > num1 and num2 > num3, print num2.
    - Otherwise, print num3.

Problem:

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

For example:

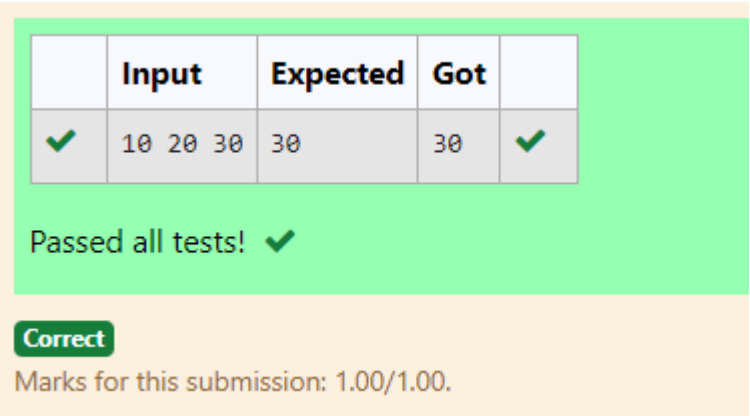
Input	Result
10 20 30	30

PROGRAM

```
#include<stdio.h>
int main()
```

```
{
    int num1,num2,num3;
    scanf("%d %d %d",&num1,&num2,&num3);
    if(num1 > num2 && num1 > num3)
    {
        printf("%d",num1);
    }
    else if(num2 > num1 && num2 >num3)
    {
        printf("%d",num2);
    }
    else
    {
        printf("%d",num3);
    }
}
```

OUTPUT:



	Input	Expected	Got	
✔	10 20 30	30	30	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

9.

Aim:

To determine whether a number is even or odd.

Algorithm:

1. Read an integer num.

2. If num % 2 == 0, print "Even".
3. Otherwise, print "Odd".

PROBLEM

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

For example:

Input	Result
12	Even
11	Odd

PROGRAM

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

OUTPUT:

	Input	Expected	Got	
✔	12	Even	Even	✔
✔	11	Odd	Odd	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

10.

Aim:
To compute the factorial of a number.

Algorithm:

1. Read an integer num.
2. Initialize fact = 1.
3. Loop from 1 to num:
    - Multiply fact by the current loop variable.
4. Print fact.

PROBLEM

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

For example:

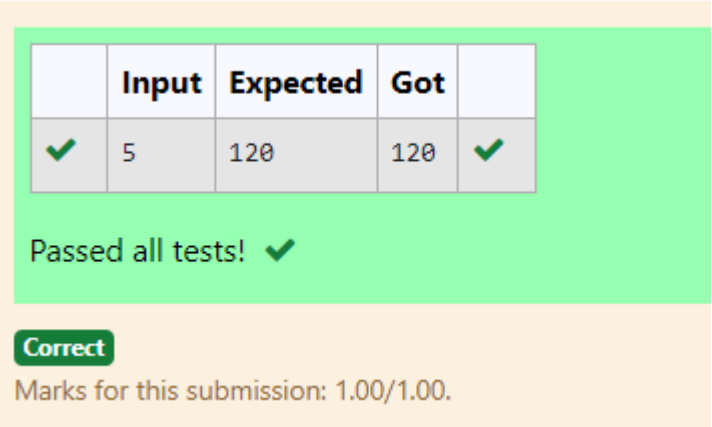
Input	Result
5	120

PROGRAM

```
#include<stdio.h>
int main()
{
   int num;
```

```
    scanf("%d",&num);
    int fact = 1;
    for(int i = 1;i <= num;i++)
    {
       fact = fact * i;
    }
    printf("%d",fact);
}
```

OUTPUT:



11.

Aim:
To calculate the sum of the first N natural numbers.

Algorithm:

1. Read an integer a.
2. Initialize N = 0.
3. Loop from 1 to a:
    - Add the current loop variable to N.
4. Print N.

PROBLEM

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

For example:

Input	Result
3	6

PROGRAM

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

OUTPUT:

	Input	Expected	Got	
✔	3	6	6	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

12.

Aim:
To find the nth Fibonacci number.

Algorithm:

1. Read an integer n.
2. If n == 0, print 0.
3. If n == 1, print 1.
4. Otherwise:
    - Initialize a = 1, b = 1.
    - Loop from 2 to n-1:
        - Compute c = a + b.
        - Update a = b and b = c.
    - Print c.

PROBLEM

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

For example:

Input	Result
0	0
1	1
4	3

PROGRAM:

```
#include<stdio.h>
int main()
{
   int n;
   scanf("%d",&n);
   if(n==0)
   {
      printf("%d",0);
```

```
    }
    else if(n == 1)
    {
        printf("%d",1);
    }
    else
    {
        int a,b,c;
        a=1,b=1;
        for(int i = 2;i<n;i++)
        {
            c=a+b;
            a=b;
            b=c;
        }
        printf("%d",c);
    }
}
```

OUTPUT:

	Input	Expected	Got	
✔	0	0	0	✔
✔	1	1	1	✔
✔	4	3	3	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

13.

Aim:
To compute the power of a number.

Algorithm:

1. Read integers a (base) and b (exponent).
2. Initialize c = 1.
3. Loop from 1 to b:
    - Multiply c by a.
4. Print c.

PROBLEM

Write a C program to find the power of integers.

input:

a b

output:

a^b value

For example:

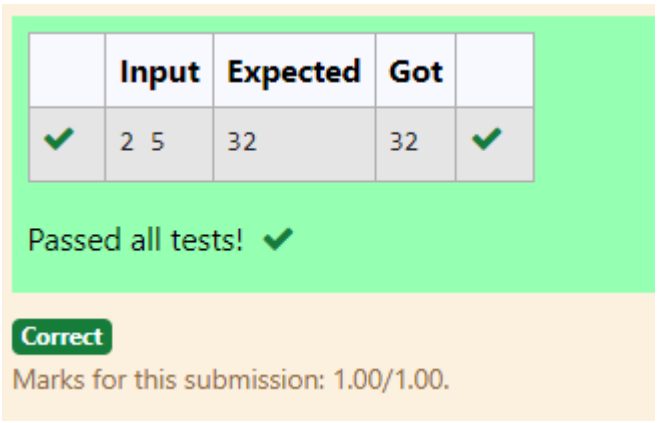
Input	Result
2 5	32

PROGRAM

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

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

OUTPUT:



	Input	Expected	Got	
✔	2 5	32	32	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

14.

Aim:
To determine whether a number is prime.

Algorithm:

1. Read an integer n.
2. Initialize c = 0.
3. Loop from 2 to n-1:
    - If n % i == 0, increment c.
4. If c > 0, print "No Prime".
5. Otherwise, print "Prime".

PROBLEM

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

For example:

Input	Result
7	Prime
9	No Prime

PROGRAM

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

OUTPUT:

	Input	Expected	Got	
✔	7	Prime	Prime	✔
✔	9	No Prime	No Prime	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

15.

Aim:
To reverse the digits of a number.

Algorithm:

1. Read an integer num.
2. Initialize rev = 0.
3. Repeat while num > 0:
    - Extract the last digit using rem = num % 10.
    - Update rev = (rev * 10) + rem.
    - Remove the last digit using num = num / 10.
4. Print rev.

PROBLEM

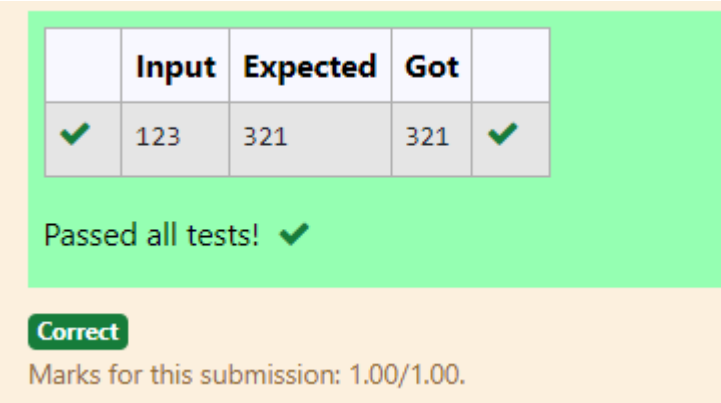
Write a C program to find the reverse of the given integer?

PROGRAM

```
#include<stdio.h>
int main()
{
   int num,rev=0,rem;
   scanf("%d",&num);
   while(num>0)
```

```
    {
        rem=num%10;
        rev=(rev*10)+rem;
        num=num/10;
    }
    printf("%d",rev);
}
```

OUTPUT:



	Input	Expected	Got	
✔	123	321	321	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.