# Basic C Programming

**PROBLEM 1**

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

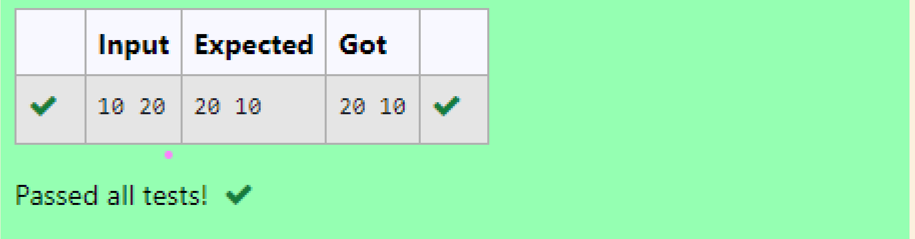
**Algorithm:**

```
DECLARE a, b, temp as

INTEGERREAD a

READ b

// Swap values of a and

btemp = a

a = b

b = temp

PRINT a, b
```

**Program:**

```
#include<stdio.h

>

int main()

{

int a

int b

;

int temp;

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

;a=b;

b=tem

p;

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

}
```

**Output:**



|  | Input | Expected | Got |  |
|---|---|---|---|---|
| ✓ | 10 20 | 20 10 | 20 10 | ✓ |

Passed all tests! ✓

## PROGRAM 2:

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

Marks in Math >= 65

Marks in Physics >= 55 [or] Total in all subjects >=180 Marks in Chemistry >= 50

**ALGORITHM:**

Step 1: Initialize m as math, p as physics, c as chemistry all as int datatype. Step 2: Input 3 numbers out of 100 from the user.

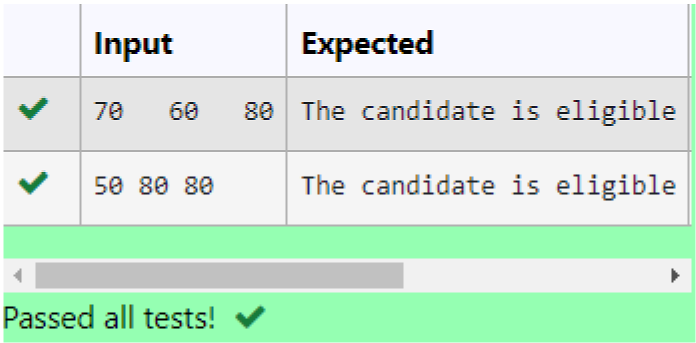
Step 3: Check if m>=65 and p>=55 and c>=50 →Then display “the candidate is eligible” Orcheck if m+p+c>=180 → Then display “the candidate is eligible”

Else → Display “the candidate is not eligible”

**PROGRAM:**

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

{

int m,p,c; scanf("%d%d%d",&m,&p,&c);

if (m>=65 && p>=55 && c>=50){ printf("The candidate is eligible");

}else if(m+p+c>=180){

printf("The candidate is eligible");

}else{

printf("The candidate is not eligible");

}}
```

**OUTPUT:**



|   | Input | Expected |
|---|---|---|
| ✓ | 70    60    80 | The candidate is eligible |
| ✓ | 50 80 80 | The candidate is eligible |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 3:

**AIM:** Malini goes to Best save hyper market to buy grocery items. Bestsave hypermarket provides 10% discount on the bill amount B whenever the bill amount B is more than Rs.2000. The bill amount B is passed as the input to the program and it must print the final amount payable by Malini.

## ALGORITHM:

Step 1: Initialize the payment and the discount as integer data types. Step 2: Take an input for payment from the user.

Step 3: Check if payment > 2000, → calculate discount as payment*0.10 and subtract it from the original payment amount.

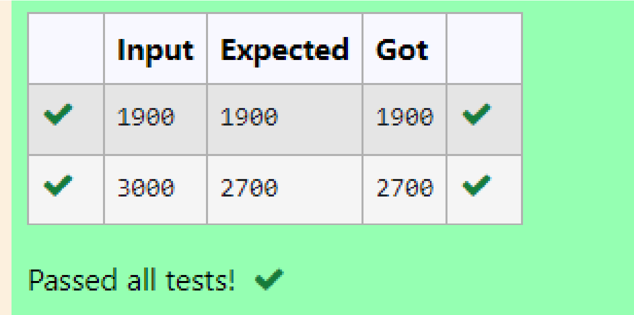
Display the new payment.

Step 4: Else → display the payment amount.

## PROGRAM:

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

{

int pay,disc; scanf("%d",&pay); if (pay>2000){

disc=pay*0.10; pay=pay-disc; printf("%d",pay);

}else{

printf("%d",pay);

}

}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 1900 | 1900 | 1900 | ✓ |
| ✓ | 3000 | 2700 | 2700 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 4:

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

## ALGORITHM:

Step 1: Initialize m and n as integer data types symbolizing the money and the number of beggars.

Step 2: Take an input from the user for the number of beggars and the money amount.

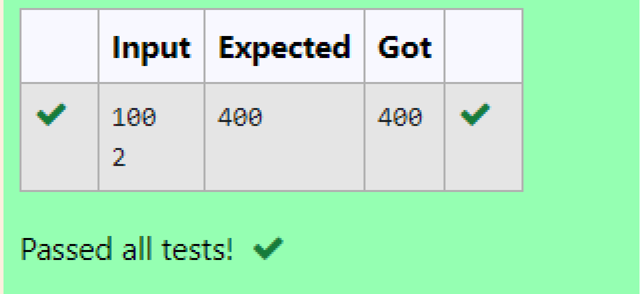
Step 3:Initialize the for loop until n, and multiply the money as money=money * n

Step 4: Outside theloop display the amount m symbolizing the money in hand.

## PROGRAM:

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

## OUTPUT:



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 100<br>2 | 400 | 400 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 5:

**AIM:** 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 [starting from Monday through Saturday] he will be awarded Rs. 200 more than the previous day as “Punctuality incentive”. Incentive for starting day is passed as input and the number of days N is also passed. The program is to calculate the “Punctuality incentive” P of the employee.

**ALGORITHM:**

Step 1: Initialize incentive i, n number of days and sum as integer datatype

Step 2: Take aninput from the user for incentive and number of days i and n.

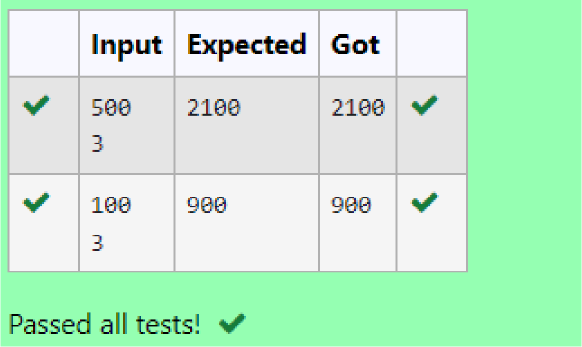
Step 3: initialize the sum as i,and initiate a for loop till n-1;

Within this for loop, calculate incentive as incentive + 200 and the sum + incentive. Step 4: Outside the loop, display the sum.

**PROGRAM:**

```
#include<stdio.h>

int main()

{

int i,n,sum;


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


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


{ i=i+200;

sum+=i;

}

printf("%d",sum);

}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 500<br>3 | 2100 | 2100 | ✓ |
| ✓ | 100<br>3 | 900 | 900 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 6:

**AIM:** Two numbers a and b are passed as the input. A number x is also passed as the input. The program must print the numbers divisible by x from b to a range inclusive of a and b.

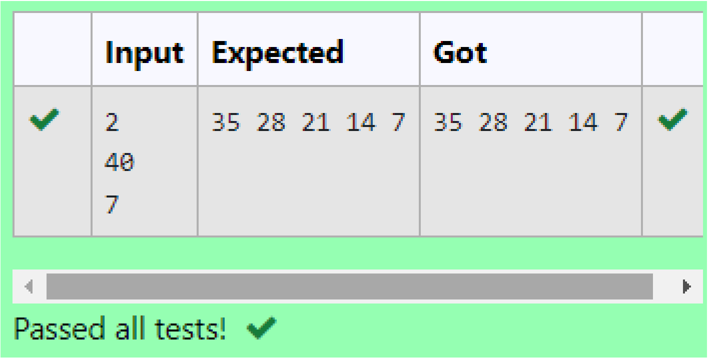
## ALGORITHM:

```
Step 1: Initialize the numbers as a, b, c as integer data types.

Step 2: Take an input for a, band c from the user.

Step 3: In a for loop, >=a, decrementing the value, Check if i%c==0, → Display the number i

Else → continue
```

## PROGRAM:

```
#include<stdio.h>

int main()

{

int a,b,c;

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

for (int i=b;i>=a;i--)

{

if(i%c==0)

{

printf("%d ",i);

}

else continue;

}

}
```

## OUTPUT:



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 2<br>40<br>7 | 35 28 21 14 7 | 35 28 21 14 7 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 7:

**AIM:** Write a program to find the quotient and remainder of the given integers.

**ALGORITHM:**

Step 1: Initialize the 2 numbers a and b.

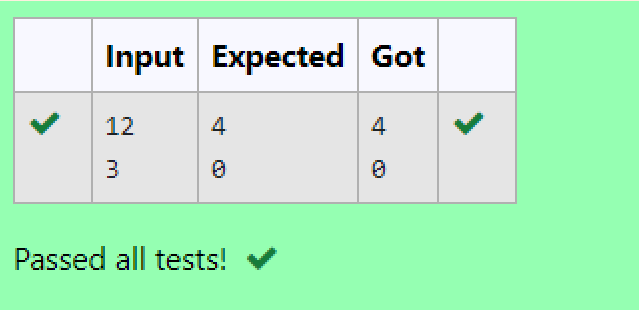
Step 2: Take an input for a and b from the user.

Step 3: Display a/b and a%b.

**PROGRAM:**

```
#include<stdio.h>

int main()

{

int a,b;


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

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

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

}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 12<br>3 | 4<br>0 | 4<br>0 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

**PROGRAM 8:**

**AIM:** Write a program to find the biggest number out of the 3 given integers.

**ALGORITHM:**

Step 1: Initialize the 3 numbers as a, b, c as integer data types.

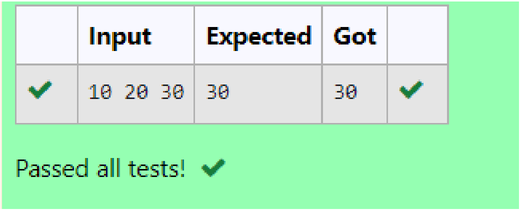
Step 2: Take an input from thea, b, c.

Step 3: Check if a>b and a>c → Display a Else check if b>a and b>c → Display b Else check ifc>a and c>b → Display c

**PROGRAM:**

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

printf("%d",c);

}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 10 20 30 | 30 | 30 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

**PROGRAM 9:**

**AIM:** Write a C program to find whether the given number is odd or even.

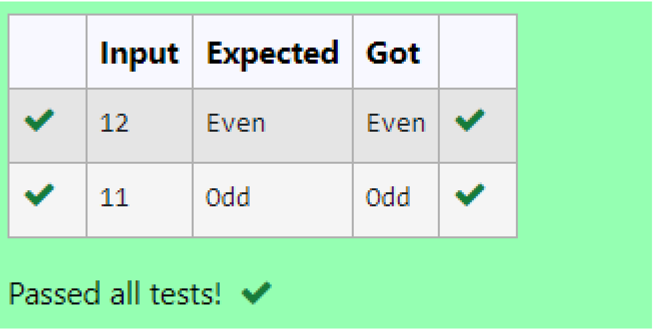
**ALGORITHM:**

```
Step 1: Initialize a number M as integer data type.

Step 2: Take an input from the user.

Step 3: Check if m%2==0 → Display even Else → Display odd.
```

**PROGRAM:**

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

**OUTPUT:**



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

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 10:

**AIM:** Write a C program to find the factorial of a number N.

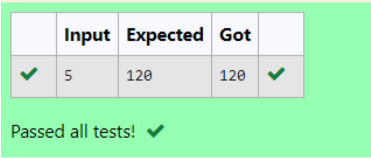
## ALGORITHM:

```
Step 1: Initialize x , i and factorial=1 as integer data type.
Step 2: Take an input for x.
Step 3: In a for loop, as i=1, and i<=x Calculate fact*=i
Step 4: Display the factorial.
```

## PROGRAM:

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

## OUTPUT:



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5 | 120 | 120 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

**PROGRAM 11:**

**AIM:** Write a C program to find the sum of first N natural.

**ALGORITHM:**

Step 1: Initialize x and sum=0 as integer data type.

Step 2: Take an input for x from the user.

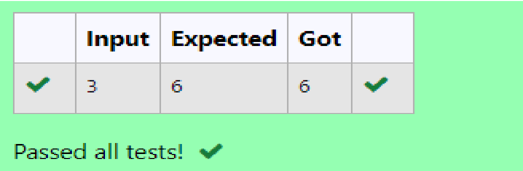
Step 3: In a for loop, i=1, i<=x, Calculate sum+=i

Step 4: Display sum.

**PROGRAM:**

```
#include<stdio.h>

int main()

{

int  x,sum=0;  scanf("%d",&x);for

(int i=1;i<=x;i++)

{

sum+=i;

}

printf("%d",sum);

}
```

**OUTPUT:**



|  | Input | Expected | Got |  |
|---|---|---|---|---|
| ✓ | 3 | 6 | 6 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 12:

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

## ALGORITHM:

Step 1: Initialize n, f0=0, f1=1, f2 and z=0, o=1 as integer data type.

Step 2: Take an input for n.

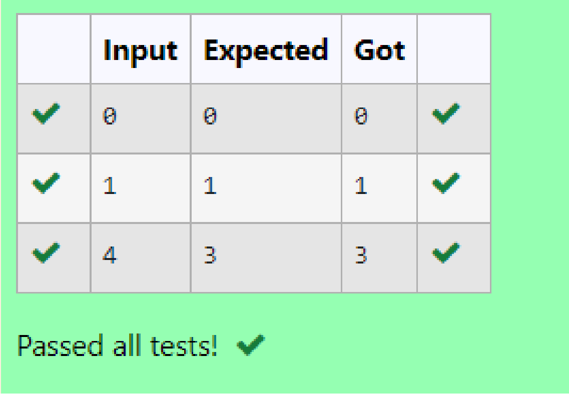
Step 3: Check if n==0, → Display z Else if n==1 → Display 0

Else calculate f2=f1+f0, f0=f1 and f1=f2 within a for loop
Step 4: Display f2.

## PROGRAM:

```
#include<stdio.h>

int main()

{

int n,f0=0,f1=1,f2,z=0,o=1;


scanf("%d",&n);


if(n==0)


 printf("%d",z);

else if(n==1)

printf("%d",o);

 else{

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

f2=f1+f0; f0=f1; f1=f2;

}
printf("%d",f2);

}}
```

**OUTPUT:**



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

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

**PROGRAM 13:**

**AIM:** Write a C program to find the powers of integers.

**ALGORITHM:**

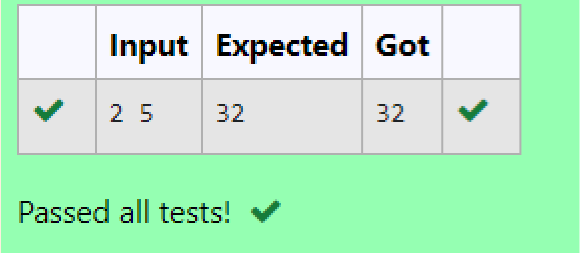
Step 1: Initialize y, x and p as integers.

Step 2: Take an input from the user for x and y. Step 3: calculate p as p=pow(x,y) and displayp.

**PROGRAM:**

```
#include<stdio.h>

#include<math.h>

int main()

{

int y,x,p; scanf("%d%d",&x,&y); p=pow(x,y); printf("%d",p);

}
```

**OUTPUT:**



|   | Input | Expected | Got |   |
|---|---|---|---|---|
| ✓ | 2 5 | 32 | 32 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

**PROGRAM 14:**

**AIM:** Write a C program to find whether the integer is prime or not.

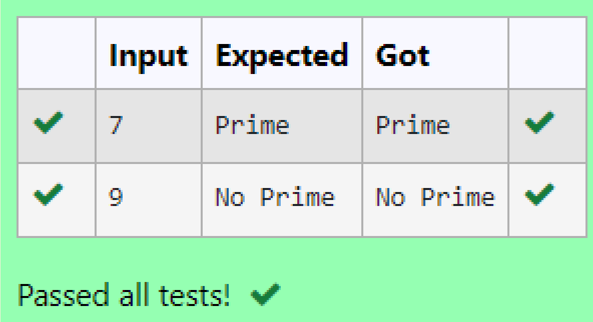
**ALGORITHM:**

```
Step 1: Initialize m as integer.
Step 2: Take an input for m.
Step 3: Check if m%2!=0 and m%3!=0 and m%5!=0 → Display prime Else → display not prime.
```

**PROGRAM:**

```
#include<stdio.h> int main()
{
int m; scanf("%d",&m);
if (m%2!=0 && m%3!=0 && m%5!=0)
{
printf("Prime");
}
else
{
printf("No Prime");
}
}
```

**OUTPUT:**



| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 7 | Prime | Prime | ✓ |
| ✓ | 9 | No Prime | No Prime | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully.

## PROGRAM 15:

**AIM:** Write a C program to find reverse of integer

## ALGORITHM:

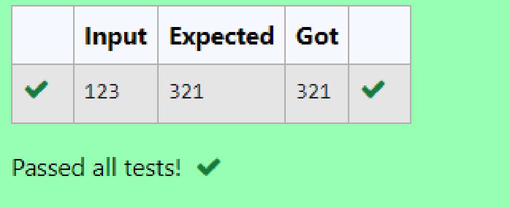
Step 1: Initialize m, rev=0 and rem as integers. Step 2: Take an input for m

Step 3: While m!=0 → rem=n%10 rev=rev*10+rem and m/=10 Step 4: Display rev

## PROGRAM:

```
#include<stdio.h>

 int main()

{

int m,rev=0,rem; scanf("%d",&m); while(m!=0)

{

rem=m%10; rev=rev*10+rem; m/=10;

}

printf("%d",rev);

}
```

## OUTPUT:



|   | Input | Expected | Got |   |
|---|---|---|---|---|
| ✓ | 123 | 321 | 321 | ✓ |

Passed all tests! ✓

**RESULT:** Thus, the program is executed successfully