# IT 161 LAB 3

Name : Snehal Keshav Nalawade

Student ID : 202151160

Date : 11/01/2022

**1)**

**Objective :** To write a C Program to find greatest in 3 numbers.

**Software used :** Online GDB Compiler and Debugger for C (IDE)

**Methodology :** This program takes three numbers as input from the user and displays the greatest one among them. If-else statements are used to determine the greatest number in this program.

## Algorithm :

Step 1 : Start

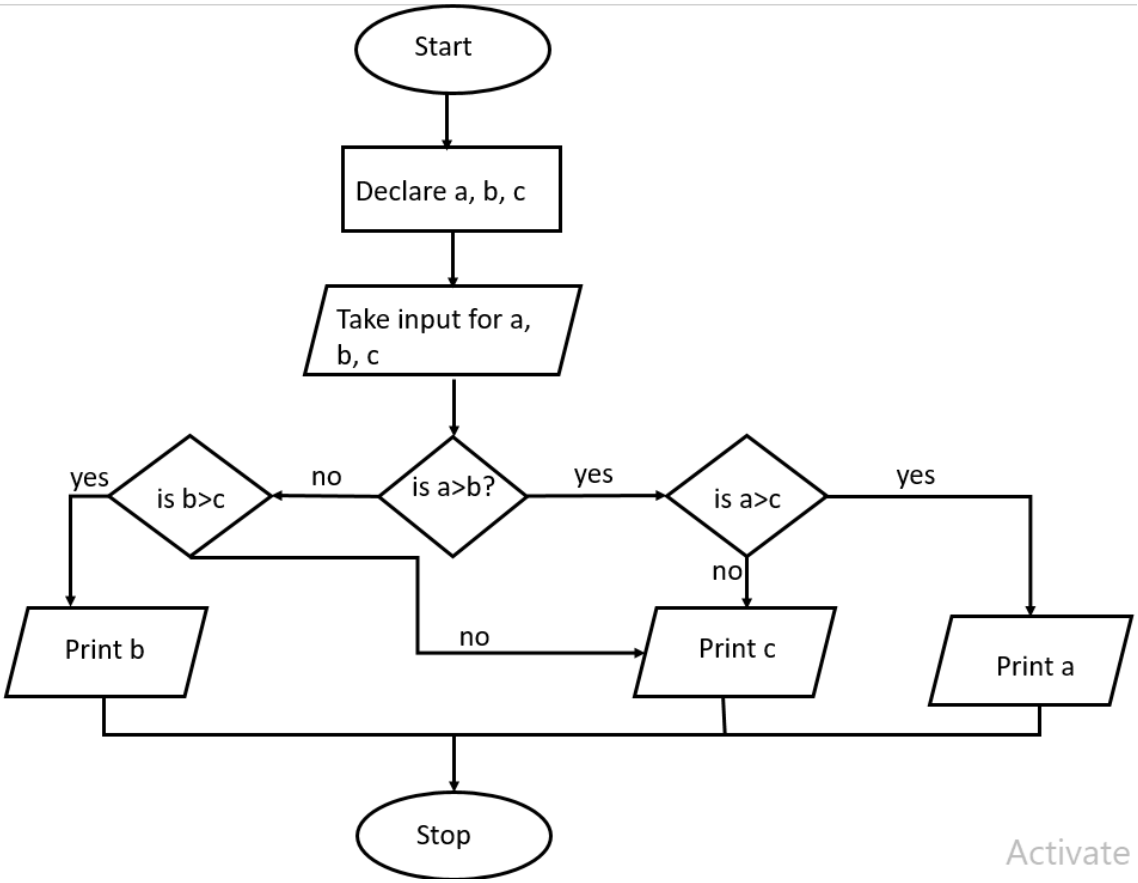
Step 2 : declare and initialize three variables a, b and c

Step 3 : take three numbers as input from the user and store in the above declared variables

Step 4 : check different cases by using if-else statements and display the greatest number among the three entered numbers

Step 5 : Stop

**Flowchart :**



## Source code :

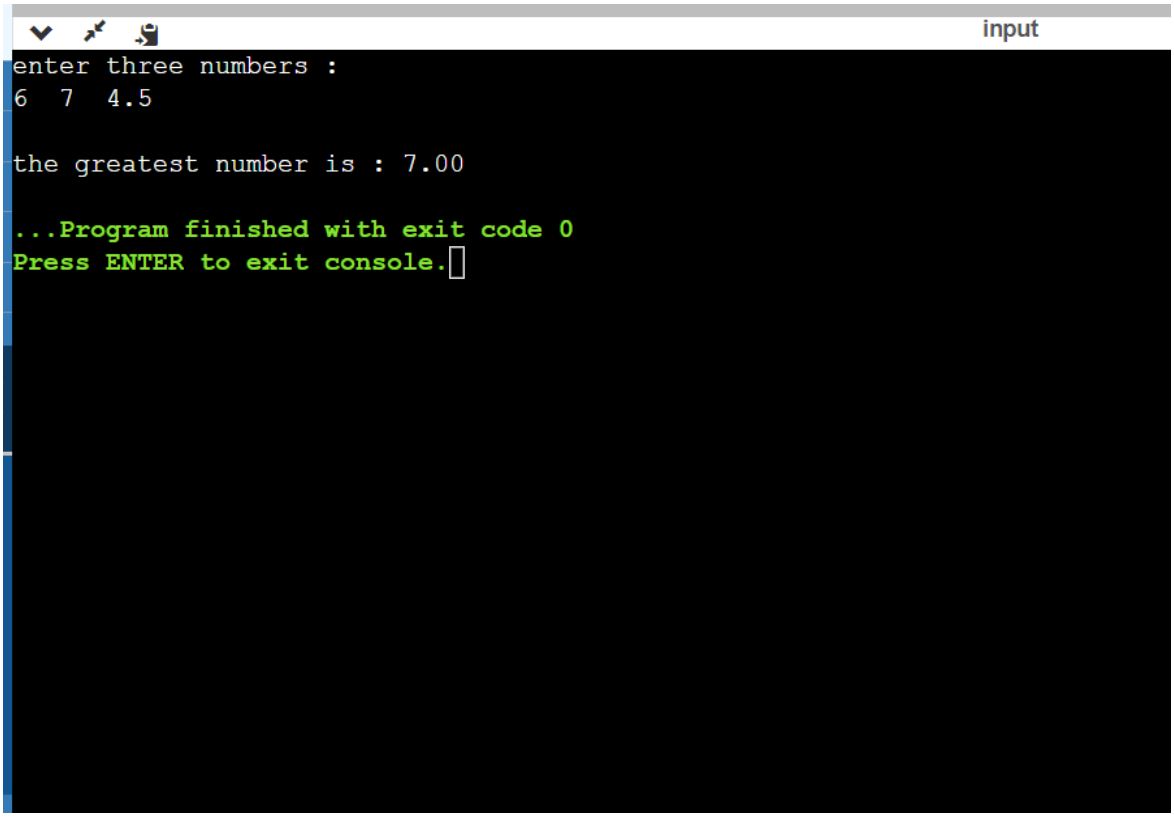
```
/* This C program is prepared by Snehal Nalawade
  Roll No : 202151160
  Date of preparation : 11/01/2022

  This program takes 3 numbers as input from the user
  and then displays the greatest number among them.
*/

#include <stdio.h>
 int main()
 {
    float a=0.0;  // declaring variable a
    float b=0.0;  // declaring variable b
    float c=0.0;  // declaring variable c
    printf("enter three numbers :\n");
    scanf("%f %f %f", &a, &b, &c);    // taking input from the user
    printf("\nthe greatest number is : ");
    if(a>b)   // start of if-else loop
    {
       if(a>c)
       printf("%.2f", a);
       else
       printf("%.2f", c);
    }
    else
    {
```

```
    if(b>c)
    printf("%.2f", b);
    else
    printf("%.2f", c);
  }
  return 0;
} // closing of the main function
```

**Sample output :**



## 2)

**Objective :** To write a C Program to show swap of two numbers
(a) By using a third variable and
(b) without using third variable.

## (a)

**Software used :** Online GDB Compiler and Debugger for C (IDE)

**Methodology :** two values are taken as input from the user. A third variable c is declared and it is used to store the value of a as entered by the user. The value of b is then assigned to a and the value stored in c is thereafter assigned to b. in this way the values of a and b are interchanged and the result is then displayed on the screen.

**Algorithm :**

Step 1 : Start

Step 2 : declare the variables a, b and c

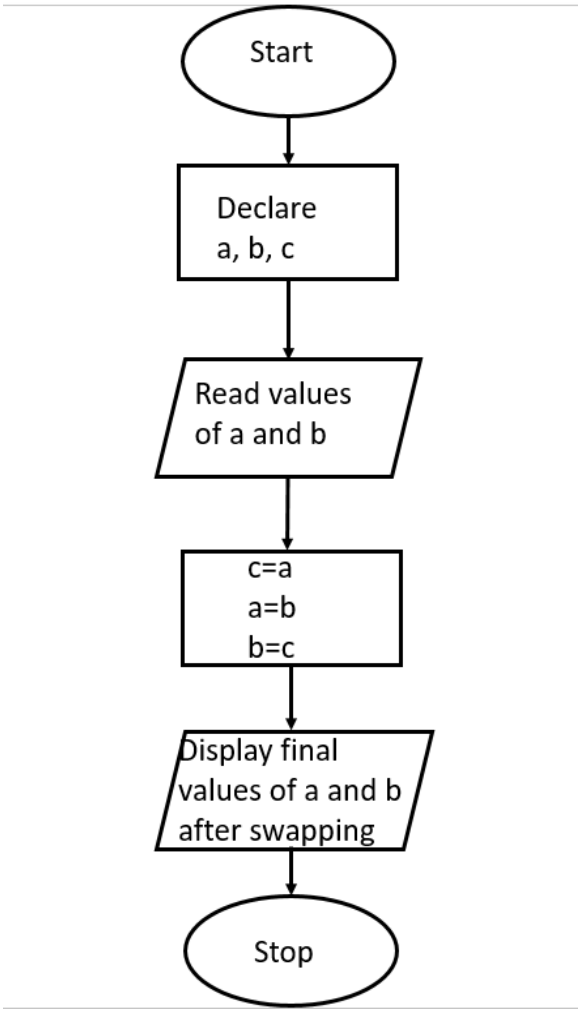
Step 3 : take values of the two numbers from the user and store in a and b

Step 4 : swap the values of a and b by using the third variable

Step 5 : display the final values of a and b after swapping

Step 6 : Stop

**Flowchart :**


Start
Declare
a, b, c
Read values
of a and b
c=a
a=b
b=c
Display final
values of a and b
after swapping
Stop

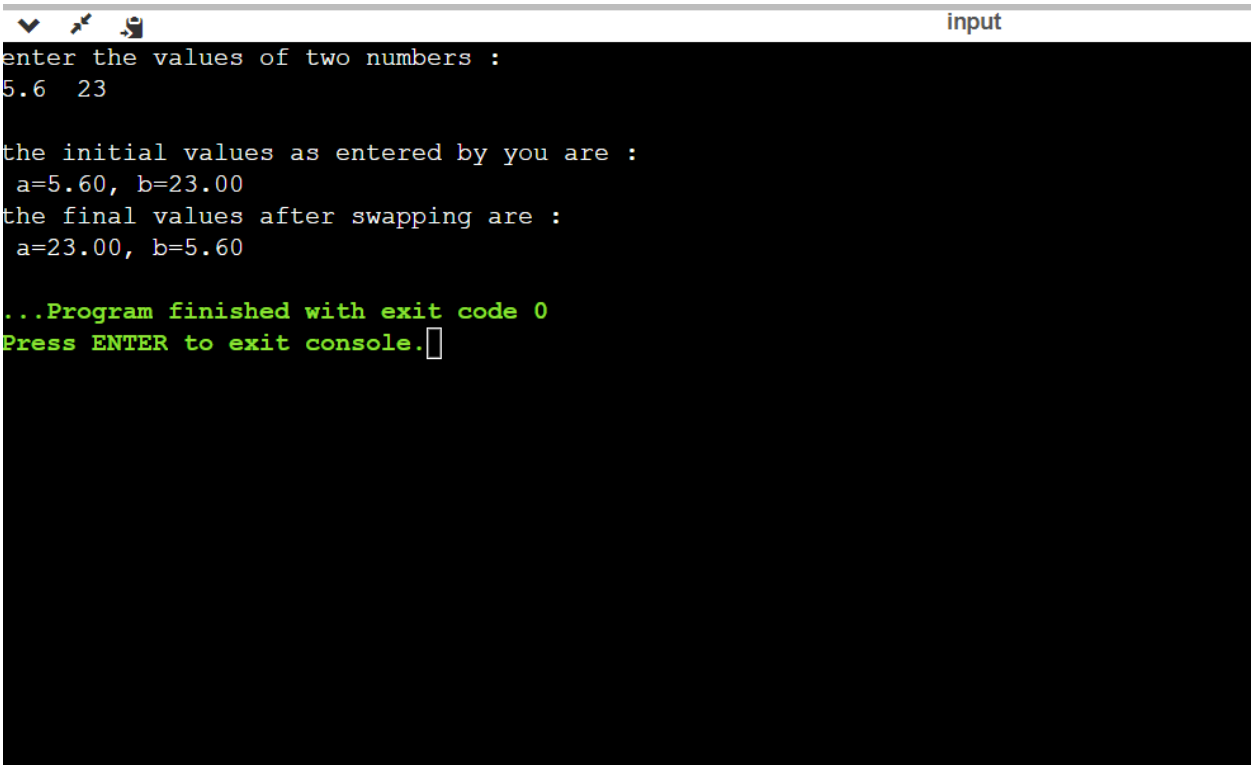
## Source code :

```
/* This C program is prepared by Snehal Nalawade
   Roll No : 202151160
   Date of preparation : 11/01/2022

   This program takes two numbers as input from the user, stores them in two distinct variables and then swaps their values.
*/

#include <stdio.h>
 int main()
{
    float a=0.0; // declaring and initialising variable a
    float b=0.0; // declaring and initialising variable b
    float c=0.0; // declaring and initialising variable c
    printf("enter the values of two numbers :\n");
    scanf("%f %f", &a, &b);  // taking input from the user
    printf("\nthe initial values as entered by you are :\n a=%.2f, b=%.2f", a, b);
    c=a;
    a=b;
    b=c;
    printf("\nthe final values after swapping are :\n a=%.2f, b=%.2f", a, b);
    return 0;
}  // closing of main function
```

**Sample output :**



## (b)

**Software used :** Online GDB Compiler and Debugger for C (IDE)

**Methodology :** two values are taken as input from the user and stored in variables a and b respectively. Further in the program, the sum of those two numbers is assigned to a and b is equated to (a-b) where a refers to the new value of a (i.e. the sum) and b refers to the original value of b which was entered by the user. Thereafter, a is equated to (a-b) where a in the RHS is equal to the sum of initial values of a and b, and b is now equal to the initial value of a. In this way the values of a and b are swapped without using a third variable. There is another way to do this as well, where we will have to do multiplication of the two values instead of addition and division instead of subtraction; rest all steps remaining the same.

## Algorithm :

Step 1 : Start

Step 2 : declare variables a and b

Step 3 : take input from the user and store in a and b

Step 4 : store the sum of a and b in a (or) store the product of and b in a

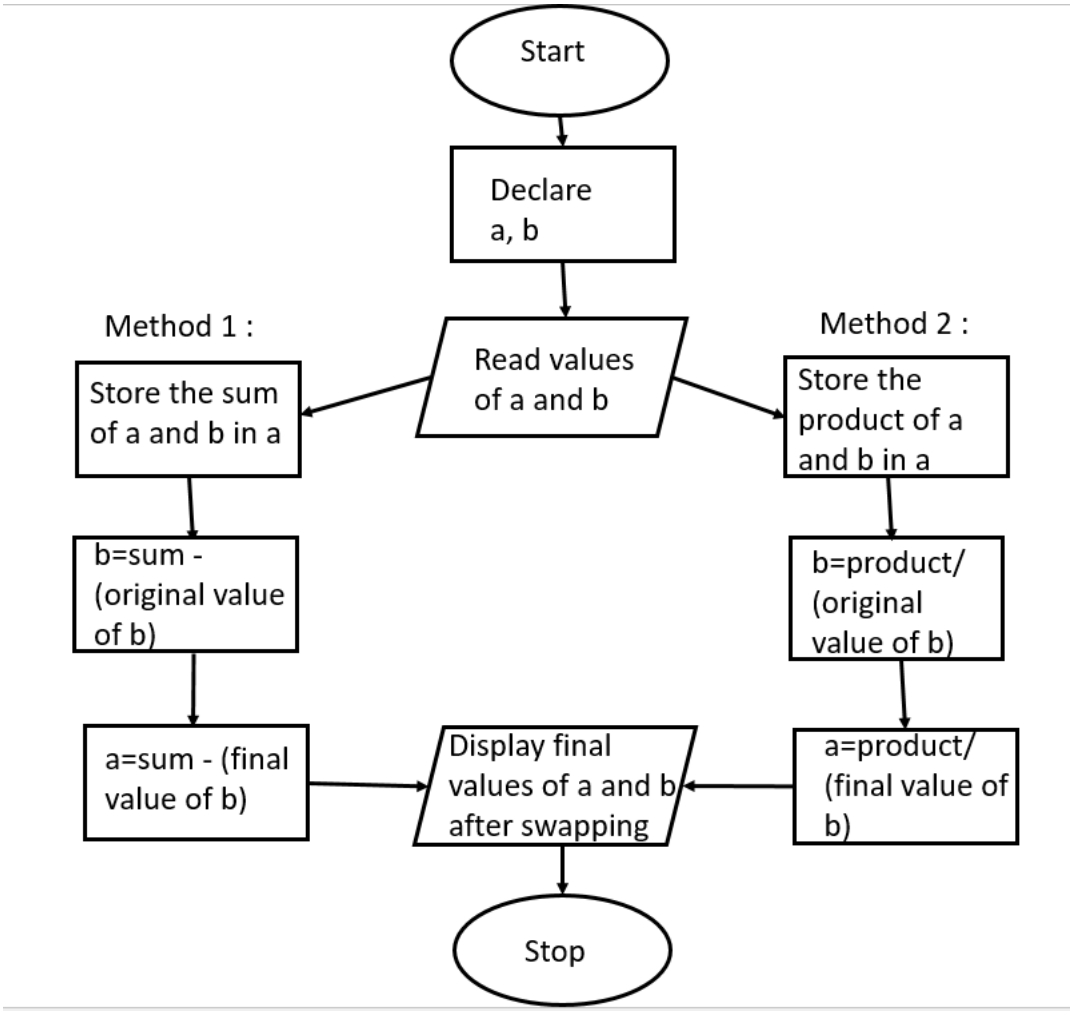
Step 5 : assign the value of (a-b) to b (or) assign the value of (a/b) to b

Step 6 : now, assign the value of (a-b) to a (or) assign the value of (a/b) to a

Step 7 : display the final swapped values of a and b

Step 8 : Stop

**Flowchart :**


Start
Declare
a, b
Method 1 :
Read values
of a and b
Method 2 :
Store the sum
of a and b in a
Store the
product of a
and b in a
b=sum -
(original value
of b)
b=product/
(original
value of b)
a=sum - (final
value of b)
Display final
values of a and b
after swapping
a=product/
(final value of
b)
Stop

## Source code :

```
/* This C program is prepared by Snehal Nalawade
  Roll No : 202151160
  Date of preparation : 11/01/2022

  This program takes two numbers as input from the user, stores them in two distinct variables
  and then swaps their values without using a third variable.
*/

#include <stdio.h>
int main()
 {
    float a=0.0, b=0.0;  // declaring and initialising variables a and b
    printf("Enter two values :\n");
    scanf("%f %f", &a, &b);  // taking two numbers as input from the user
    printf("\nthe initial values as entered by you are :\n a=%.2f, b=%.2f", a, b);
    a=a+b;  // storing the sum of a and b in variable a
    b=a-b;
    a=a-b;
    printf("\nthe final values after swapping are :\n a=%.2f, b=%.2f", a, b);
    return 0;
 }  // closing of the main function
```

(OR)

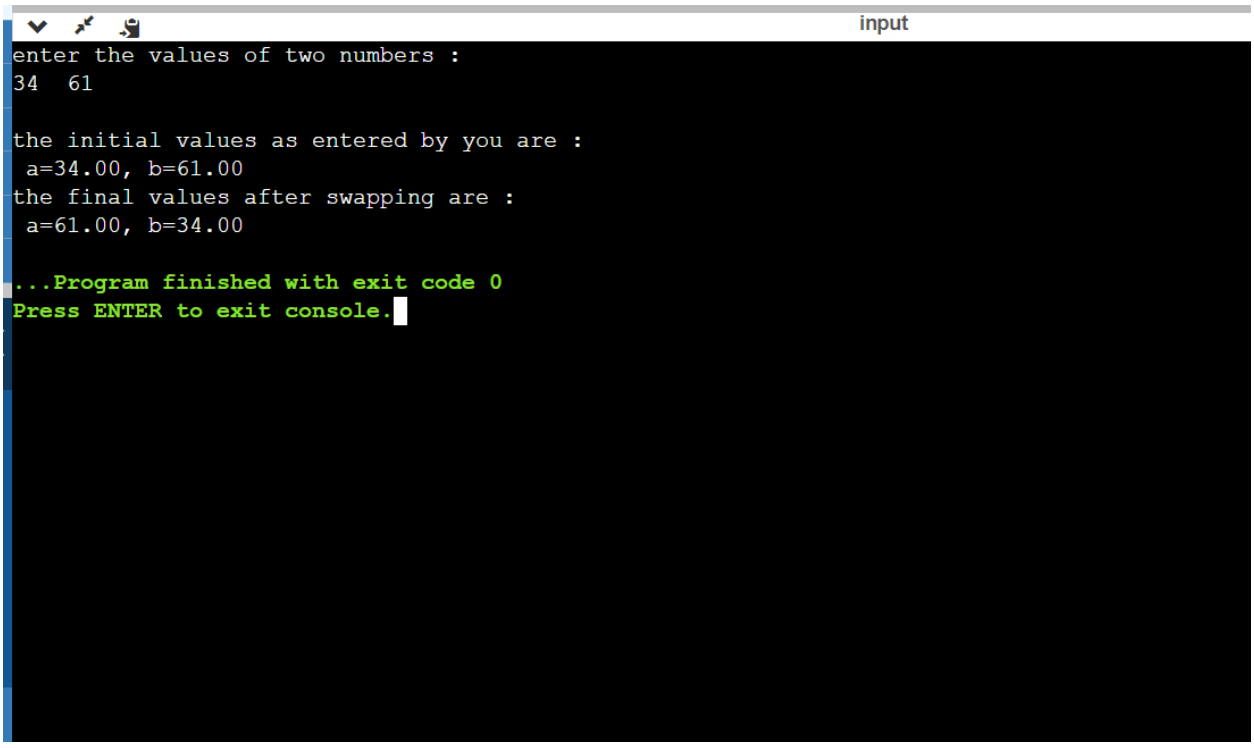
```
/* This C program is prepared by Snehal Nalawade
  Roll No : 202151160
  Date of preparation : 11/01/2022

  This program takes two numbers as input from the user, stores them in two distinct variables
  and then swaps their values without using a third variable.
*/

#include <stdio.h>
```

```
int main()
 {
    float a=0.0, b=0.0;  // declaring and initialising variables a and b
    printf("Enter two values :\n");
    scanf("%f %f", &a, &b);  // taking two numbers as input from the user
    printf("\nthe initial values as entered by you are :\n a=%.2f, b=%.2f", a, b);
    a=a*b;  // storing the product of a and b in variable a
    b=a/b;
    a=a/b;
    printf("\nthe final values after swapping are :\n a=%.2f, b=%.2f", a, b);
    return 0;
 } // closing of the main function
```

## Sample output :



**3)**

**Objective :** To write a C Program to reverse the digits of a given number (assuming that the given number has a max of 3 digits. The program should first find the number of digits in the given integer)

**Software used :** Online GDB Compiler and Debugger for C (IDE)

**Methodology :** the program first takes input from the user and stores it in a variable named var. then if-else statements are used to find out the number of digits in the entered value (this program is written under the assumption that the value entered by the user will have a maximum of 3 digits; so, if the entered value has more than three digits, it will be considered as an invalid input). After the number of digits are known, if-else statement and arithmetic operators are used to separate the digits of the entered value and then add them in such a way that we get the reversed number. The result is then displayed on the screen.

**Algorithm :**

Step 1 : Start

Step 2 : declare variables var, rvar, numdig, rem

Step 3 : take input from the user and store it in var

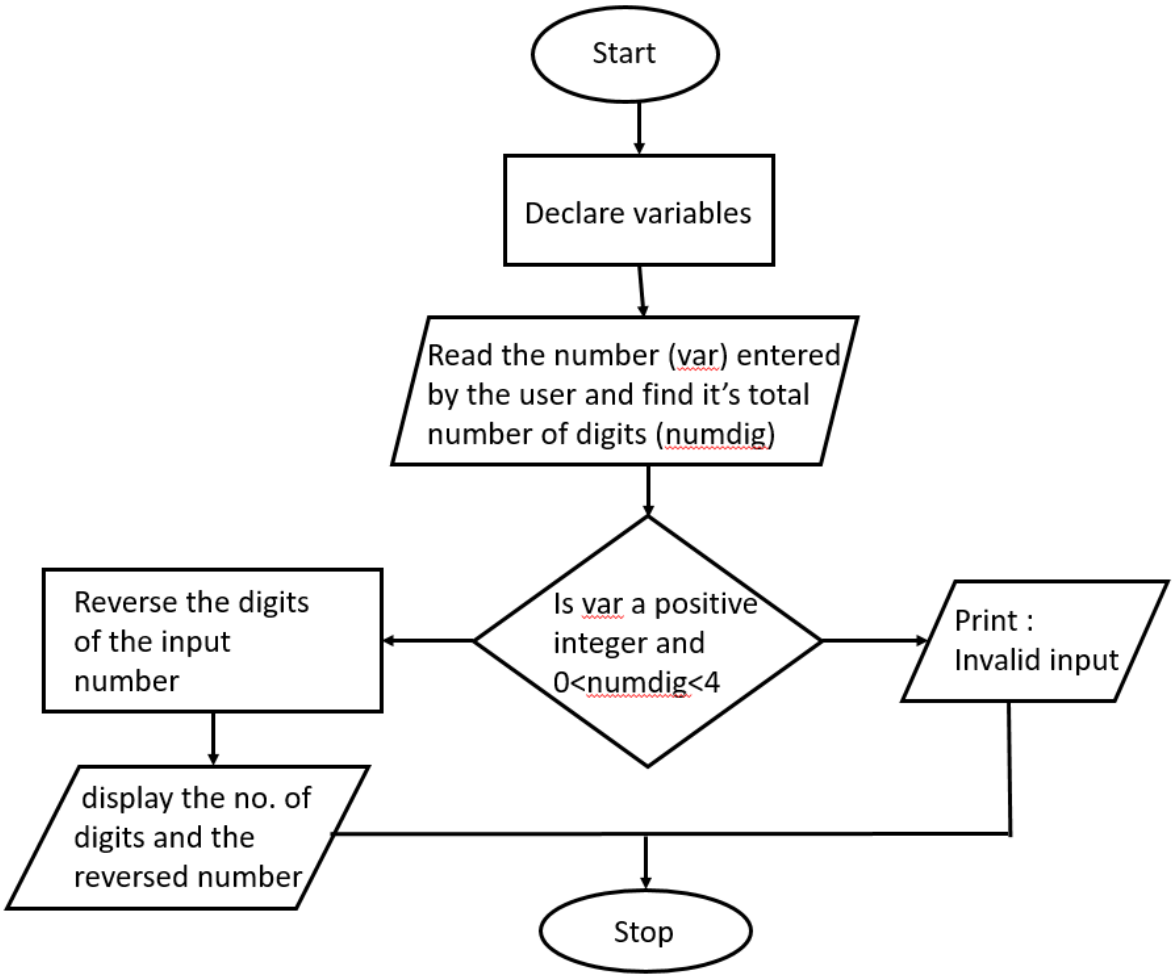
Step 4 : find the number of digits and store in numdig (if numdig > 3, then 'Invalid input' will be displayed on the screen)

Step 5 : reverse the number by using if-else statement and arithmetic operations

Step 6 : display the result

Step 7 : Stop

## Flowchart :


Start
Declare variables
Read the number (var) entered by the user and find it's total number of digits (numdig)
Is var a positive integer and 0<numdig<4
Reverse the digits of the input number
Print : Invalid input
display the no. of digits and the reversed number
Stop

## Source code :

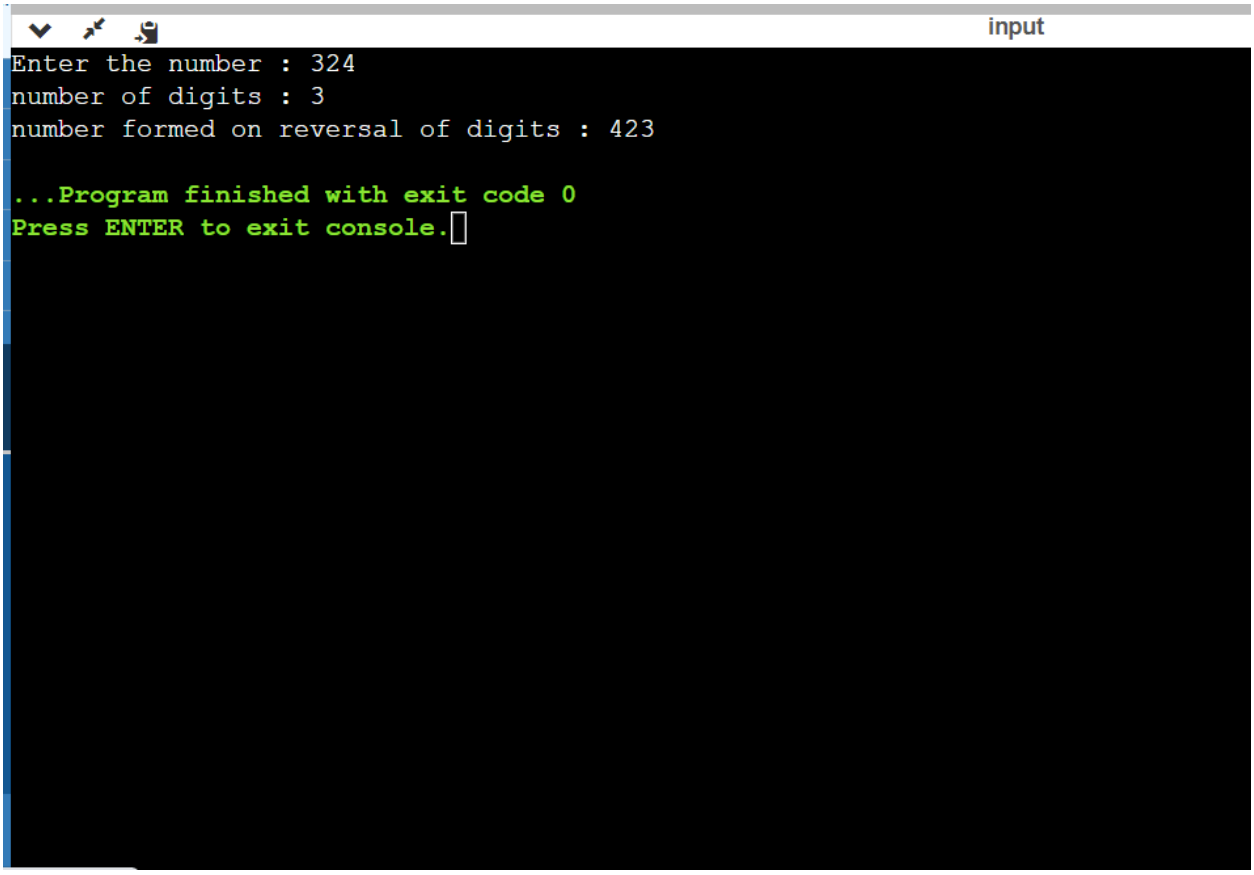
```
/* This C program is prepared by Snehal Nalawade
  Roll No : 202151160
  Date of preparation : 11/01/2022

  This program takes an integer number of upto 3 digits as input from the user
  and displays the number formed by reversing the digits of the entered value.
*/

#include <stdio.h>
 int main()
 {
    int var=0;
    int numdig=0, rvar=0, rem=0;   // declaring variables
    printf("Enter the number : ");
    scanf("%d", &var);   // taking input from the user
    if(var>=0 && var<10)   // checking condition for if statement
    {
       numdig = 1;
       rvar = var;
       printf("number of digits : %d\n", numdig);
       printf("number formed on reversal of digits : %d", rvar);
    }
    else if(var>=10 && var<100)
    {
```

```
    numdig = 2;
    rem = var%10;
    rvar = (rem*10)+(var/10);
    printf("number of digits : %d\n", numdig);
    printf("number formed on reversal of digits : %d", rvar);
  }
  else if(var>=100 && var<1000)
  {
    numdig = 3;
    rem = var%10;
    int var1 = var/10;
    rvar = (rem*100)+(var1%10*10)+(var1/10);
    printf("number of digits : %d\n", numdig);
    printf("number formed on reversal of digits : %d", rvar);
  }
  else
  printf("Invalid input : Please enter a positive integer with a maximum of 3 digits only");
  return 0;
} // closing of the main function
```

**Sample output :**




# Thank you