# For While Do while Break Continue Assignment

1. WAP to read a number n and to display the cumulative sum of factorial of all numbers upto n . (use for or while)

Input: 4

Ouput: 4!+3!+2!+1! = 32

A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

1. Write a program to accept “N” integers from the user. “N” also has to be taken from the user. Take the count of +ve numbers, -ve numbers and 0’s.

However the program should not accept a non-integer value. If a non-integer value is entered, used must be asked to re-enter.

[Hint:

* 1. Use the return value of scanf to find out whether the user has entered integer or not.
  2. You also will have to clear the input buffer before taking the next input.

For clearing the input buffer, use one of the following approaches

* while (getchar() != '\n'); // keep reading till newline and discard the characters
* scanf(“%\*s”); // read and discard one string

]

A computer screen shot of a program code

Description automatically generated

A computer screen shot of numbers and letters

Description automatically generated

1. Write a program to continuously read a string of maximum length 80 chars, End the program if string is END, else convert to upper case, display and continue. (use while)

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer code

Description automatically generated

1. Refer the program “value\_out\_of\_domain.c”. Try to run the program with a large value say 255. Check the output? Is it correct? Fix the issue observed.

What improvements do you suggest?

Ans) The factorial of 255 is a high value which cannot be stored in int data type which holds only 4 bytes with a range of -2,147,483,648 to 2,147,483,647. So to store a large values we need to define an array with large size like array[1000] that holds the value.

1. Refer the code below. It does not output anything. Fix it.

#include <stdio.h>

int main()

{

int x = 5;

while (x > 0);

{

printf( "Value of x :%d \n", x);

x--;

}

return 0;

}

Ans: while (x>) here semicolon should not be placed. The semicolon after the while (x > 0) in the original code was causing the loop to do nothing inside. By removing it, the loop now properly executes the block inside the curly braces {} as long as x > 0.

1. Analyse the code, identify the issues

#include <stdio.h>

int main()

{

float cnt = 0, num = 1000;

do

{

printf ("\n%d\n%d", num,cnt);

num /= cnt;

} while (cnt --); /\* End of while \*/

return 0;

}

1. **Correct the format specifiers** to %f for float.

2. **Avoid division by zero** by setting a non-zero value for cnt.

3. **Modify the loop logic** to ensure cnt behaves as intended.