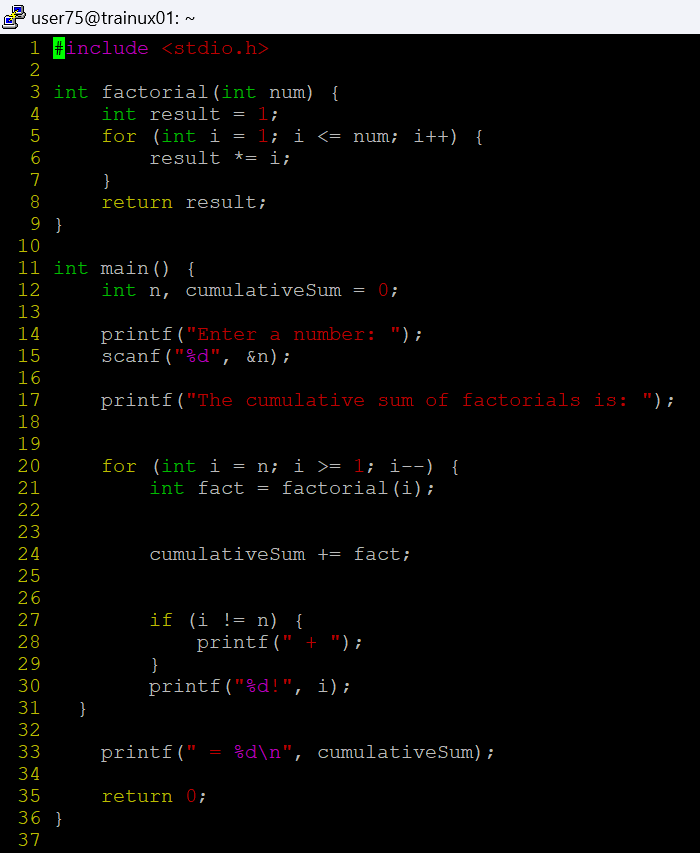
**For While Do while Break Continue Assignment**

**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 black background with numbers and symbols

Description automatically generated**

**2. 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:**

**a. Use the return value of scanf to find out whether the user has entered integer or not.**

**b. You also will have to clear the input buffer before taking the next input.**

**For clearing the input buffer, use one of the following approachesØ while (getchar() != '\n'); // keep reading till newline and discard the characterØ scanf(“%\*s”); // read and discard one stringA screen shot of a computer program

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**3. 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 program

Description automatically generated**

**4. 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?**

The factorial of large numbers like 255 results in an extremely large value that cannot be stored in standard data types like int or long due to their limited size. For instance, 255! (factorial of 255) is a 508-digit number, which far exceeds the range of any built-in integer type in C.

using arrays allows to handle very large numbers, like the factorial of 255, which would otherwise be impossible with basic data types in C.

**5. 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;**

**}**

To fix this, we should remove the semicolon after the while loop so that the block { printf("Value of x : %d \n", x); x--; } becomes part of the while loop and gets executed on each iteration.

**6. 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;**

**}**

The issue with the original code is that it uses the format specifier %d to print the values of cnt and num, both are declared as float types. The format specifier %d is used for integers.

so using it for float variables will lead to incorrect behavior and can cause issues.