# RECURSIVE FUNCTION

1. WAP to calculate the maximum stack depth of a recursive call to a function. (For eg a factorial function ).

A:

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer code

Description automatically generated

1. What is tail recursion? Why is it important? Give an example

A: Tail recursion is a special type of recursion where the recursive call is the **last** operation in the function. In other words, the function performs its work and then directly calls itself, without needing to perform any additional computation after the recursive call returns.

void print(int n)

{

if (n < 0)

return;

printf("%d ", n);

// The last executed statement is recursive call

print(n - 1);

}

Compilers usually execute recursive procedures by using a stack. This stack consists of all the pertinent information, including the parameter values, for each recursive call. When a procedure is called, its information is pushed onto a stack, and when the function terminates the information is popped out of the stack.

Thus for the non-tail-recursive functions, the stack depth (maximum amount of stack space used at any time during compilation) is more.