Assignment no - 9

Title - Find factorial of given number

Problem statement - Write X86 ALP to find factorial of given integer number on a command line by using recursion. Explicit stack manipulation is expected in the code.

Objective - To understand how to use stack segment for recursion

Outcome - student will study recursion using stack in ALP.

S/w and H/w requirements - core 2 duo i3/i5, i7, OS Linux 32/64 bit, editor : gedit, Assembler : NASM, Debugger : GDB

Concept related theory -

PUSH → Push operand onto the stack

push decrements the stack pointer by 2 if the operand size - attribute of the instruction is 16 bit, otherwise it decrements stack pointer by 4. Push the places the operand on the new top of the stack, which is pointed to by stack pointer.

The 80386 push esp instruction pushes the value of esp as it existed before the instruction. This differs from the 80386 where the push SP pushes the new value.

POP → Pop word from the stack

- Pop replaces the previous contents of the memory, the register or the segment register operand with the word on top of 80387 stack, addressed by SS:SP (address size attribute of 16 bit) or SS:ESP (address size, attribute of 32 bit)

Stack pointer SP is incremented by 2 for an operand size of 16 bit or by 4. For an operand size of 32 bit

Factorial -

Product of all positive integers less than or equal to given positive integer

e.g $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

Algorithm -

1. Start
2. Accept the number from user
3. Convert the number into hex
4. Compare accepted number with 1. If it is equal to 1 go to step 5 else push the number on stack and decrement the no. and goto step 4.
5. Pop content of stack and multiply with number.
6. Repeat step until stack become empty
7. Convert the number from hex to ascii
8. Print the number
9. End.

Testcases

	Testcase	Expected O/P	Actual O/P	Result
①	./a.out 05	78H	78H	Pass
②	./a.out 04	18H	18H	Pass
③	./a.out 00	Factorial is 0001	Factorial is 0001	Pass
④	./a.out 06	02D0	02D0	Pass

Conclusion - In this way we studied the use of stack and recursively find the factorial of number with stack operations (push and pop)