Pranav Velamakanni
Assignment 3
07/26/2015

Basic outline of the implementation

1. Added main function.

2. Added Error handling for input command line arguments.

3. Added function to print the formula in C.

4. Added Factorial Function in assembly.

5. Added nCr funtion in assembly.

6. Added the Overflow error handling to nCr.

7. Added Overflow error handling to print formula function in C.

8. Tested for various possible inputs

9. Added Makefile

Runtime Analysis

Runtime of this depends on input n passed alongside the command line arguments.

For input n. print_formula function runs for n+1 times which is O(n). For each loop body. It calculates nCr +constant work.

So basically for this formula problem Runtime T(n) = O(n).T(nCr)

nCr calculation involves Caculation of n!,n-r! and r!. Which is Theta(n)

So overall runtime for this problem is O(n^2).

Space Analysis

Since it only allocates constant no. of variables each time it runs. Space Requirement = Theta(n)

Challenges encountered while implementation

1. First problem was to run code for 32 bit architecture on 64 bit machine. I followed 2 different techniques. a) Using Virtual Machine of 32 bit architecture. b) use of flag -m32 to compile for 32 bit machine on 64 bit machine.

2. Next problem was implementation of assembly language factorial. My issue was that I was getting Segmentation fault as I was not saving the base pointer and registers. To resolve the issue I had to review some of these concepts from Wiki's GAS assembly language tutorial.

3. I was getting floating point error as I was not clearing %edx before passing idivl instruction.