Marcus Lomi
NetID : Mal403
CS211: Computer Architecture
Programming Assignment 2

I. Calc:

A. Description: Calc.c is designed to take in command line arguments for addition subtraction and multiplication of numbers input in decimal, octal, binary, or hexadecimal formats.

B. Command Line Arguments

1. Example execution: ./calc + d10 b11 d
2. Format: <operation> <number1> <number2> <output base>
3. Operation accepts either +, -, or * to signify addition, subtraction or multiplication of numbers. NOTE: * must be surrounded by double quotes to be read correctly as part of the command line argument.
4. Number: must be input in the format that signifies the base with proper following digits. Example: xf for hexadecimal representation of 15, or b11 for binary representation of 2.
5. Output base must be one character long, upper or lower case, and must be either x, b, d or o, for Hexadecimal, binary, decimal or octal format output.

C. Design:

1. Firstly, correct format is checked at the very beginning of the program so no issues arise later. Verifying the exact length of the operator and output base arguments are completed in constant time. Checks for the correct input format of the individual number arguments are in linear time however; O(n+m) where n is the length of number 1 and m is the length of number 2. This is because successful inputs will have been traversed completely to check for invalid characters. All of the successful strings, regardless of base are run through the toDecimalConversion method which uses simple modulo operations to convert the characters into decimal integers saved by the program. These decimal integers are then added, subtracted, or multiplied and stored in a single int. From there, the decimal int is converted to a string of the output base type. For simplicity purposes, negative integers are made positive before being converted into strings and the negative sign is appended at the end in the printout. The program cannot handle numbers which go above the 32-bit max range, overflows do occur.