Using MASM

The process of editing, assembling, linking and executing assembly language programs is very similar to the same processes for Visual C++. See the accompanying description for how to use MASM with either Visual Studio C++ or Visual Studio C++ Express.

You will first need to copy the Irvine files (www.kipirvine.com/asm/examples) and install them. Then follow the steps on the handout to get the Project_Sample file to execute.

Program 0

Next, change the content of the sample program as follows and get it to run.

```
TITLE Hello program

; Programmer:
; Description:
;
; Date:

INCLUDE Irvine32.inc

      .data
myMessage   BYTE  "Hello World!",0dh,0ah,0

      .code
Main  PROC
      call  Clrscr

      mov   edx,OFFSET myMessage
      call  WriteString

      exit
main  ENDP

END   main
```

CSCI/CMPE 2333 -- Program 1
due October 27, 2016

Write a program in assembly language that will

- read in four integers
  - you will need to call the routine ReadInt four times
    - one number per line
  - this routine reads a value from the keyboard and stores it in EAX

```
call    ReadInt
mov     Num, eax
```

- if the integers are A, B, C, and D, compute
  - A - B + C - D
    - (Note: C is a reserved word so you cannot use it as an identifier!)
- print out the integers and the result
  - use the routine WriteInt which will display the contents of the register EAX on the screen as a signed decimal

```
mov     eax, Num
call    WriteInt
```

Use the following set of integers (in decimal):

42, -5, -23, 35

Use only the instructions covered so far. (Don't use looping!)

Hand in:
1) A copy of the source files (not the listing file!)
2) A copy of the output from your program
   (copy the output screen as demonstrated in class
   - Right click on icon in upper left corner of output
   - Select edit -> Mark
   - Highlight the selection
   - Press Enter key
   - Go to bottom of your program
   - Right click and Paste)

CSCI/CMPE 2333 Program Guidelines

The documentation of your programs should meet the following guidelines:

a) Should have a title section that includes
   - Name
   - Course and section
   - Program title and date
   - Description of problem

b) Each module should have an introductory comment

c) Use descriptive label names when possible

d) At least 50% of lines of assembly programs should have informative comments (unless otherwise instructed)

Grading of the programs will be on the following basis:

- 25% Documentation
- 55% Organized, efficient and correct logic
- 20% Organization and correctness of output, with sufficient output to cover all cases

NOTE: You are expected to create your own programs. Copying, collusion, or other forms of cheating will not be tolerated. Programs that are not your own work will earn no credit.

Computer unavailability due to down time or lack of sufficient machines should be anticipated and planned for. Expect Murphy's Law to be in effect!