CPSC 240: Computer Organization and Assembly Language
Assignment 02, Fall Semester 2023

CWID:__________________ Name:__________________________

1. Download the “CPSC-240 Assignment02.docx” document.
2. Design the "addition.asm" program, and use assembly language to realize the function of the following C++ instructions.

    unsigned short num1 = 50000;
    unsigned short num2 = 40000;
    unsigned int sum = 0;
    sum = int(num1 + num2);
3. Assemble the "addition.asm" file and link the "addition.o" file to get the "addition" executable file.
4. Run the "addition" file with the DDD debugger to display the simulation results of num1 and num2, as well as the simulation results of sum.
5. Insert source code (addition.asm) and simulation results (DDD debugger window) of the memory (num1, num2, and sum) in the document. Use calculator or hand calculation to verify the simulation results.
6. Design the "subtraction.asm" program, and use assembly language to realize the function of the following C++ instructions.

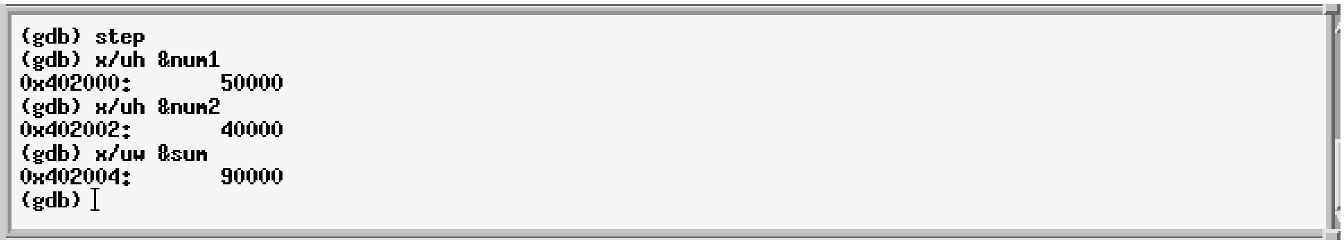
    signed short num1 = 20000;
    signed short num2 = 30000;
    signed int dif = 0;
    dif = int(num1 - num2);
7. Assemble the "subtraction.asm" file and link the "subtraction.o" file to get the "subtraction" executable file.
8. Run the "subtraction" file with the DDD debugger to display the simulation results of num1 and num2, as well as the simulation results of diff.
9. Insert source code (subtraction.asm) and simulation results (DDD debugger window) of the memory (num1, num2, and dif) in the document. Use calculator or hand calculation to verify the simulation results.
10. Save the file in pdf format and submit the pdf file to Canvas before 23:59 pm on 09/13/2023.

[Insert addition.asm here]

```
; addition.asm
; unsigned short num1 = 50000;
; unsigned short num2 = 40000;
; unsigned int sum = 0;
; sum = int(num1 + num2);
```

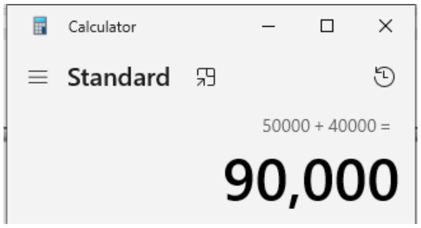
```
section .data
       num1    dw     50000                       ;num1 = 50000
       num2    dw     40000                       ;num2 = 40000
       sum     dd     0                           ;sum = 0

section .text
       global _start

_start:
       mov     dx, 0                              ;dx = 0
       mov     ax, word[num1]                     ;ax = num1
       add     ax, word[num2]                     ;ax = ax + num2
       adc     dx, 0                              ;dx = dx + 0 + CF
       mov     word[sum+0], ax                    ;sum+0 = ax
       mov     word[sum+2], dx                    ;sum+2 = dx
                                                  ;sum = dx:ax
       Mov     rax, 60                            ;terminate excuting process
       mov     rdi, 0                             ;exit status
       syscall                                    ;calling system services
```

[Insert addition simulation result here]



```
(gdb) step
(gdb) x/uh &num1
0x402000:       50000
(gdb) x/uh &num2
0x402002:       40000
(gdb) x/uw &sum
0x402004:       90000
(gdb)
```

[Insert addition simulation result verification here]



50000 + 40000 =

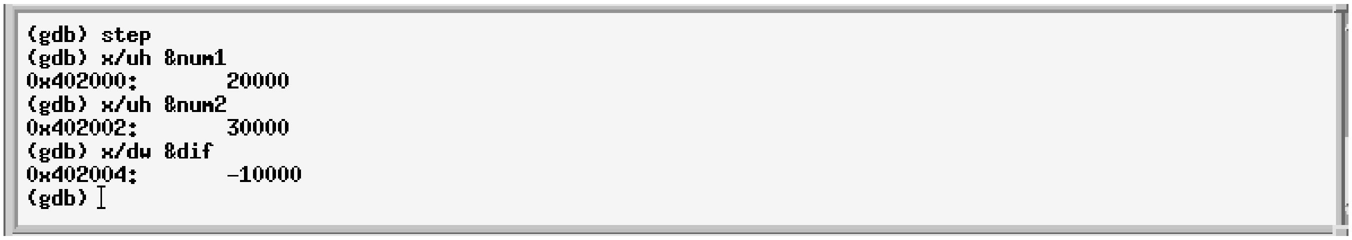
90,000

[Insert subtraction.asm here]

```
;subtraction.asm
; signed short num1 = 20000;
; signed short num2 = 30000;
; signed int sum = 0;
; dif = int(num1 - num2);
```

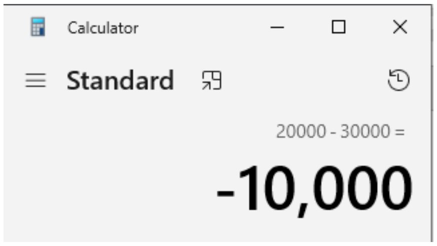
```
section .data
        num1    dw      20000                           ;num1 = 20000
        num2    dw      30000                           ;num2 = 30000
        dif     dd      0                               ;dif1 = 0

section .text
        global _start

_start:
        mov     dx, 0                                   ;dx = 0
        mov     ax, word[num1]                          ;ax = num1
        sub     ax, word[num2]                          ;ax = ax - num2, CF = 1
        sbb     dx, 0                                   ;dx = dx - 0 - CF
        mov     word[dif+0], ax                         ;dif+0 = ax
        mov     word[dif+2], dx                         ;dif+2 = dx
                                                        ;dif = dx:ax
        Mov     rax, 60                                 ;terminate excuting process
        Mov     rdi, 0                                  ;exit status
        Syscall                                         ;calling system services
```

[Insert subtraction simulation result here]



```
(gdb) step
(gdb) x/uh &num1
0x402000:       20000
(gdb) x/uh &num2
0x402002:       30000
(gdb) x/dw &dif
0x402004:       -10000
(gdb)
```

[Insert subtraction simulation result verification here]



20000 - 30000 =

-10,000