Computer Science Department
California State University, Fullerton

CPSC 240-01 Computer Organization and Assembly Language
Final Exam
1:00 PM to 2:15 PM
Tuesday, May 16, 2023

Student Name: Andrew Saldana

Last 4 digits of ID: 0327

Note:

- University regulations on academic honesty will be strictly enforced.
- You have 75 minutes to complete this Quiz.
- Open books, slides and sample programs.
- Turn off or turn vibration your cell phone.
- Use YASM assembler for the program design.
- Copy and paste your assembly source code and Terminal Emulator window to the end of the word file and save it in pdf or docx format.
- Submit you pdf or docx file to Canvas before the deadline.
  NOTE: Email submissions will not be graded.
- Any content submitted after the due date will be regarded as a make-up quiz.

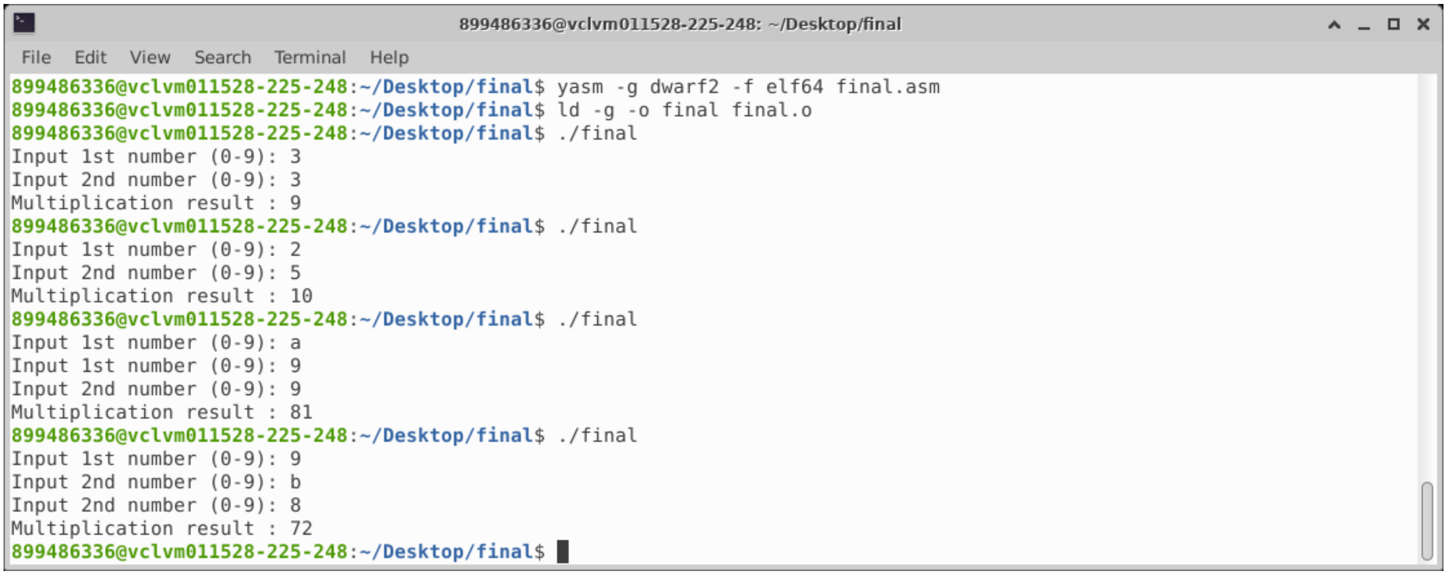
Final Exam

1. Download the “CPSC-240-01 Final Exam.docx” document.
2. Use x86-64 assembly language to implement the following C/C++ program.

```
#begin define print(addr, n)
  rax = 1;
  rdi = 1;
  rsi = addr of string;
  rdx = n;
  syscall;
#end
#begin define scan(addr, n)
  rax = 1;
  rdi = 1;
  rsi = addr of buffer;
  rdx = n;
  syscall;
#end
char num1, num2, result;
char buf[2];
char msg1[24] = "Input 1st number (0~9): ";
char msg2[24] = "Input 2nd number (0~9): ";
char msg3[24] = "Multiplication result : ";
char ascii[3] = "00\n";
void main() {
  rbx = &msg1;
  call toNumber(rbx);
  num1 = al;
  rbx = &msg2;
  call toNumber(rbx);
  num2 = al;

  al = num1;
  bl = num2;
  call multiplication();
  result = al;

  di = short(result);
  call toAscii();
  cout << msg3;
  if(result < 10)
      cout << ascii+1;
  else
      cout << ascii;
}
void toNumber(char[] message) {
  do {
    cout << message;
    cin >> buf;
  } while(buf >= '0' && buf <= '9');
  al = atoi(buf);
}
void multiplication() {
  ax = al * bl;
}
void toAscii() {
  ascii = itoa(result);
}
```

3. After assembling and linking, run the executable file to display the simulation results in the Terminal Emulator window as the following example.
4. Insert source code and the simulation results (Terminal Emulator window) to the bottom of the document.
5. Save the file in pdf or docx format and submit the pdf or docx file to Canvas before the deadline.
6. Deadline is 2:15 pm on 05/16/2023.

Simulation result example:


899486336@vclvm011528-225-248: ~/Desktop/final
File Edit View Search Terminal Help

```
899486336@vclvm011528-225-248:~/Desktop/final$ yasm -g dwarf2 -f elf64 final.asm
899486336@vclvm011528-225-248:~/Desktop/final$ ld -g -o final final.o
899486336@vclvm011528-225-248:~/Desktop/final$ ./final
Input 1st number (0-9): 3
Input 2nd number (0-9): 3
Multiplication result : 9
899486336@vclvm011528-225-248:~/Desktop/final$ ./final
Input 1st number (0-9): 2
Input 2nd number (0-9): 5
Multiplication result : 10
899486336@vclvm011528-225-248:~/Desktop/final$ ./final
Input 1st number (0-9): a
Input 1st number (0-9): 9
Input 2nd number (0-9): 9
Multiplication result : 81
899486336@vclvm011528-225-248:~/Desktop/final$ ./final
Input 1st number (0-9): 9
Input 2nd number (0-9): b
Input 2nd number (0-9): 8
Multiplication result : 72
899486336@vclvm011528-225-248:~/Desktop/final$ 
```

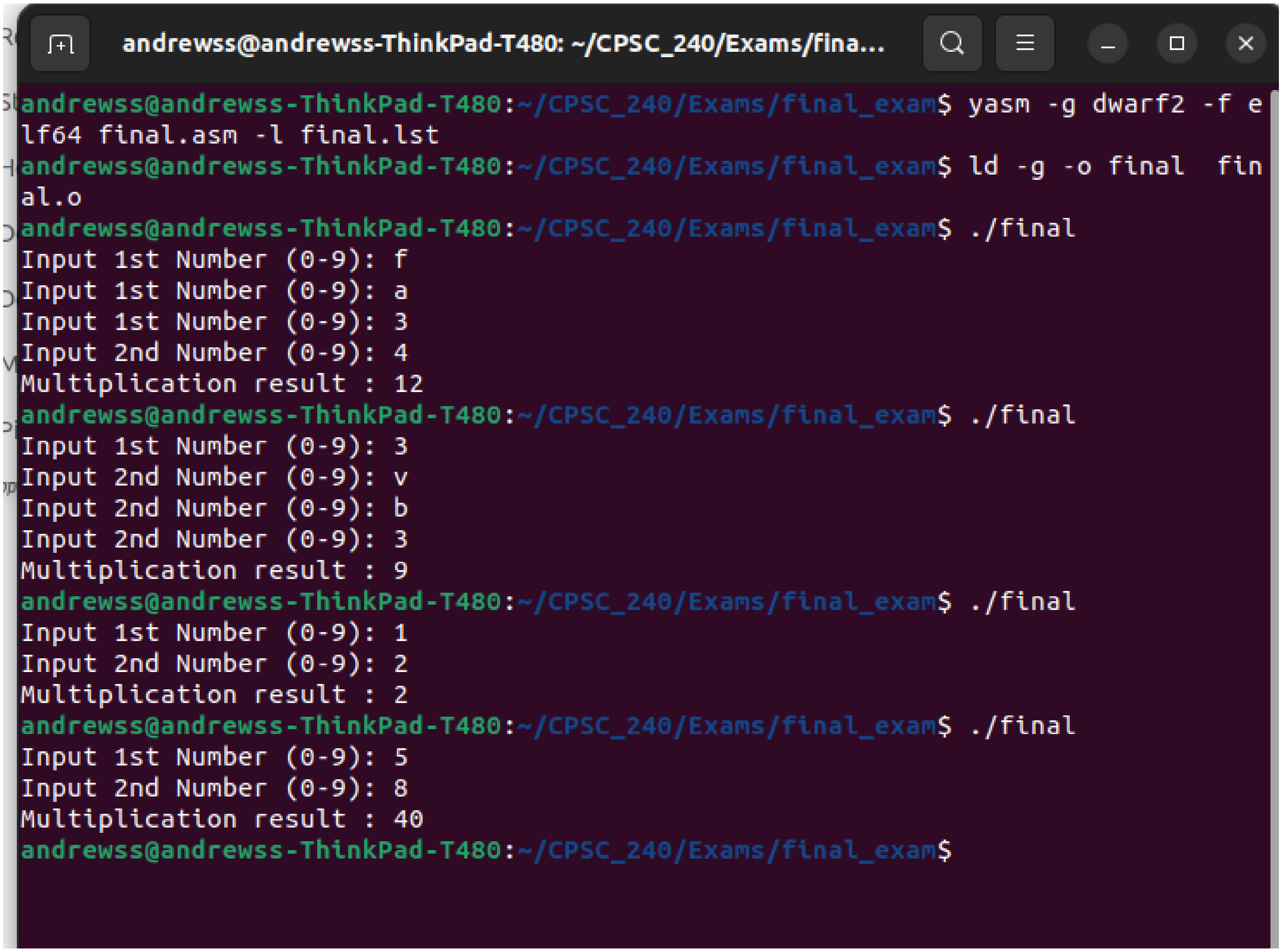
[Attach your assembly source code here:]

```
%macro     print      2
        mov      rax, 1                           ;SYS_write
        mov      rdi, 1                           ;standard output device
        mov      rsi, %1                          ;output string address
        mov      rdx, %2                          ;number of character
        syscall                              ;calling system services
%endmacro

%macro     scan  2
        mov      rax, 0                           ;SYS_read
        mov      rdi, 0                           ;standard input device
        mov      rsi, %1                          ;input buffer address
        mov      rdx, %2                          ;number of character
        syscall                              ;calling system services
%endmacro

section    .data
```

```
     msg1  db    "Input 1st Number (0-9): "
     msg2  db    "Input 2nd Number (0-9): "
     msg3  db    "Multiplication result : "
     ascii db    "00", 10

section    .bss
     buffer     resb 2
     num1  resb 1
     num2  resb 1
     result     resb 1

section    .text
     global     _start

_start:
     mov   rbx, msg1
     call  toNumber
     mov   byte[num1], al

     mov   rbx, msg2
     call  toNumber
     mov   byte[num2], al

     mov   al, byte[num1]
     mov   bl, byte[num2]
     call  multiplication
     mov   byte[result], al

     mov   di, word[result]
     call  toAscii

     cmp   word[result], 10
     print msg3, 24

     jl    one_dig

     print ascii, 3
     jmp   end
```

```
    one_dig:
        print ascii+1, 2
        jmp   end

    end:
        mov   rax, 60
        mov   rdi, 0
        syscall

;*************************To Number Function***********************************

toNumber:
    print rbx, 24
    scan  buffer, 2

    cmp   byte[buffer], 0x30
    jl    toNumber
    cmp   byte[buffer], 0x39
    jg    toNumber

    mov   cl, byte[buffer]
    and   cl, 0x0f
    mov   al, cl
    ret

;***********************MUltiplication
Function*********************************

multiplication:
    movzx ax, al
    mul   bl
    ret

;***************************To                                             Ascii
Function***********************************

toAscii:
    mov   ax, di                                ;ax = result
    mov   bx, 10                                ;bx = 10
    mov   rcx, 1           ;rcx  =  1  (because  possibility  of  being  2  digit
number)
next2:
    mov   dx, 0                         ;dx = 0
    div   bx                            ;dx=(dx:ax)%10, ax=(dx:ax)/10
    add      byte[ascii+rcx], dl                    ;ascii+rcx = al + 30h
    dec   cx                            ;cx--
    cmp   cx, 0                         ;compare cx and 0
    jge   next2                         ;if (cx>=0) jump to next2
    ret
```

[Attach Terminal Emulator window here:]



```
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ yasm -g dwarf2 -f e
lf64 final.asm -l final.lst
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ ld -g -o final  fin
al.o
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ ./final
Input 1st Number (0-9): f
Input 1st Number (0-9): a
Input 1st Number (0-9): 3
Input 2nd Number (0-9): 4
Multiplication result : 12
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ ./final
Input 1st Number (0-9): 3
Input 2nd Number (0-9): v
Input 2nd Number (0-9): b
Input 2nd Number (0-9): 3
Multiplication result : 9
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ ./final
Input 1st Number (0-9): 1
Input 2nd Number (0-9): 2
Multiplication result : 2
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$ ./final
Input 1st Number (0-9): 5
Input 2nd Number (0-9): 8
Multiplication result : 40
andrewss@andrewss-ThinkPad-T480:~/CPSC_240/Exams/final_exam$
```