**Computer Science Department**

**California State University, Fullerton**

CPSC 240-01/02 Computer Organization and Assembly Language

Quiz 01

12:00 pm to 1:15 pm

Monday, October 7, 2024

Student Name:

Last 4 digits of ID:

**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 to assemble the source code.
* Use “ld” linker to link the object code
* Use “ddd” debugger to simulate the executable code.
* Each student can only submit solution once, and secondary submissions will not be graded. If you have submitting problems, please inform your instructor before you leave the classroom.
* Any content submitted after the due date will be regarded as a make-up quiz.

Quiz 01

1. Download the “CPSC-240-01 Quiz 01.docx” document.
2. Convert the following C/C++ variable declarations and arithmetic operations to x86-64 assembly language. Use the “yasm” assembler to assemble the program, the “ld” linker to link the object code, and the “ddd” debugger to simulate the executable code.   
   NOTE: variable sizes and program functions should be equivalent to C/C++ instructions.

signed short num1 = +30000; //16-bit signed variable

signed short num2 = +20000; //16-bit signed variable

signed short num3 = -3333; //16-bit signed variable

signed int sum = 0 //32-bit signed variable

signed short quo = 0; //16-bit signed variable

signed short rmd = 0; //16-bit signed variable

sum = int(num1 + num2);

quo = sum / num3;

rmd = sum % num3;

1. After assembling and linking, run the DDD/GDB debugger to display the simulation results of the decimal values of num1, num2, num3, sum, quo, and rmd in GDB panel before terminate program.
2. Insert source code and the simulation results (GDB panel) to the bottom of the document.
3. Save the file in pdf or docx format and submit the pdf or docx file to Canvas before the deadline.
4. Deadline is 1:15 pm on 10/07/2024.

[Copy and paste your assembly source code here:]

[Attach GDB window with all memory data here:]