

**CS2100 Computer Organization**  
**AY2024/25 Semester I**  
**Assignment 1**  
**Deadline for submission:**  
**Monday, 16 September 2024**  
**1PM**

Instructions

1. There are FIVE (5) questions in this assignment, totaling THIRTY-SEVEN (37) marks. Please do all parts of every question. Marks are indicated against each part.
2. An additional 3 marks is awarded for putting in your name, student ID and tutorial group number in your answers and for submitting in PDF format with the correct naming convention (see below). Thus, the total will be FORTY (40) marks. If you fail to do any of these, you will lose the 3 marks.
3. This assignment is due on **Monday, 16 September 2024, \*1\*pm. 5 marks will be deducted automatically from late submission for every 10 minutes pass the deadline. No submission or a submission after Monday, 16 September 2024, 2 pm will automatically get ZERO for this assignment, regardless of how hard you've worked on it.**
4. **There is no limit to the number of times you can submit before the deadline. The version closest to the deadline will be the one that is graded.**
5. Part 1: This Assignment is split into two parts. Questions 2, 3 and 5 are to be attempted in the respective "Quizzes" on Canvas. The questions are randomly assigned and the answers are auto-graded via string matching. You have 5 attempts each and the answers will not be revealed till after the deadline. Due to the use of string matching, you will need to adhere strictly to the format of each answer specified in the question.
6. Part 2: The C program solution to Questions 1 and 4, i.e., **Assg1Q1ans.c** and **Assg1Q4ans.c**. **Put these into a subfolder called "AxxxxxxxY" and ZIP this folder into a zip file named AxxxxxxxY.zip, and submit that to the Assignment folder on Canvas for your TUTORIAL group, named "Assignment 1 – Submission for Tutorial Group Txx". Your ZIP file should therefore unzip into a subfolder "AxxxxxxxY" on the grader's computer. In other words, after unzipping your submission AxxxxxxxY.zip, we should see a subfolder "AxxxxxxxY" with exactly the two C programs (with no other folder or files): Assg1Q1ans.c and Assg1Q4ans.c. You will forfeit the 2 marks if you do not do this, or if you fail to fill in your name, student ID and tutorial group number in expected places in the C files. In addition, if you do not submit your C files, you will not receive marks for the respective questions.**

Sorry but due to the sheer size of the class, we have to be draconian about this. And submitting to the respective Assignment 1 folder for your tutorial group is the only way we can organise the submission effectively – so please do not get this wrong.

7. You should do these assignments on your own. **Do not discuss** the assignment questions with others. The NUS Policy on Plagiarism shall apply strictly. The use of AI is also prohibited because the exercises will help you deepen your skills.

8. Please use the “Assignment 1” topic in the SETS forum for this module for clarifications.

---

### Question 1. (20 MARKS)

In this question, you will complete a C program to do (naïve) addition in 1’s complement. You are provided with a skeleton file ‘**Assg1Q1.c.skeleton**’ and ‘**main.c.skeleton**’ in the same Canvas folder as this question sheet. Do the following steps:

1. Create a new folder for the assignment that bears your student ID. We have to deal with a massive number of submissions and if you do not follow the instructions carefully, we will have to penalize you for making our life difficult.
2. Download ‘**Assg1Q1.c.skeleton**’ from the same Canvas folder as this question into newly created folder, and rename it ‘**Assg1Q1ans.c**’.
3. Do the same for ‘**main.c.skeleton**’, and rename it ‘**main.c**’.
4. You are *not* to modify **main.c** *other than* changing the two **#DEFINE** at its start to indicate your student number and name (as strings within quotes). These will appear in the output.
5. Your work will need to be done in **Assg1Q1ans.c**.
  - a. Modify the three lines of comments (lines 9 to 11) at the beginning of **Assg1Q1ans.c** to identify yourself.
  - b. Complete the C code that has been marked as “<Your code here...>” in **Assg1Q1ans.c**
6. Compile your code with:

```
$ cc -o Assg1Q1ans Assg1Q1ans.c main.c
```
7. Run it with:

```
$ ./Assg1Q1ans
```
8. Here is what a correct output might look like:

```

$ ./Assg1Q1ans
CS2100 Assignment 1 Question 1
-----
One's complement addition submission by:
Student name: <Your name here within quote, i.e., as a string>
Student number: <Your student ID here as a C string>
=====
How many bits do you want to do this in (must be [2, 32])? 16

Input your first number (as signed decimal): 123

Input your second number (as signed decimal): -64

Here we go...

X:  0000000001111011 (123)
Y:  1111111110111111 (-64) +
-----
C1: 1111111111111110
Z1: 000000000111010 (58)
      0000000000000001 +      <-- due to carry out
-----
C2: 0000000000000000
Z2: 000000000111011 (59) [FINAL ANSWER in one's complement]

```

We will grade your answer using hidden test cases. In order to do that, we need you to stick strictly to the API given. Hence do not change the given function prototypes or main.c other than where you are suppose to identify yourself. Your submission for this question shall only be **Assg1Q1ans.c** in your folder.

Important note: Your submission must at least compile correctly without warnings or errors using the default GCC flags. The correctness of the program will be assessed separately.

As stated in the introduction, this is an individual effort. Do not discuss with others.

### Question 2 (3 MARKS) and Question 3 (4 MARKS)

These are to be answered in Canvas. They are available via Canvas->CS2100->Assignments->"Assignment 1 Question 2" and Canvas->CS2100->Assignments->"Assignment 1 Question 3", respectively. The questions are randomized and so you will get a different question if you make another attempt. You have 5 attempts, and the answers will be revealed after the deadline.

#### Question 4. (5 MARKS)

For the following MIPS implementation for **func**, write down the equivalent C code for **func**. In your C code, use the following variable to register assignments: **\$v0** mapped to integer variable **count**, **\$t0** mapped to integer variable **i**. You will need to figure out the correct function prototype for **func**. You may not need to map all registers to C variables. For example, registers that are just used to store temporary intermediate results may not need to be mapped to a C variable. Note that the code only uses integer variables and constants (if any).

Submit your answer as **Assg1Q4ans.c** in your answer ZIP file. Note that one of the grading criteria shall be whether your submitted **Assg1Q4ans.c** compiles and runs correctly.

```
func:      li      $t0, 32
           add     $v0, $zero, $zero
           j       $L3

$L6:      addi    $v0, $v0, 1
           addi    $t0, $t0, -1

$L7:      beq     $t0, $zero, $L8
           sll     $a0, $a0, 1

$L3:      slti    $t1, $a0, 0
           beq     $t1, $zero, $L6
           addi    $t0, $t0, -1
           j       $L7

$L8:      jr      $ra
```

```
void func (int a) {
    int i = 32; count = 0;
    while (i > 0) {
        if (a >= 0) count++;
        i--;
    }
    a *= 2;
}
```

#### Question 5. (5 MARKS)

These are to be answered in Canvas. They are available via Canvas->CS2100->Assignments->"Assignment 1 Question 5". The questions are randomized and so you will get a different question if you make another attempt. You have 5 attempts, and the answers will be revealed after the deadline.

=== END OF ASSIGNMENT ===