

COMP3350

Lab 2

Important Notes:

- There will be three submitted **assembly language files (.asm file)**. The **code** must be accompanied by adequate comments. Writing only the code could lead to **zero** credits.
- The submitted **lab report** should be a **single PDF file** unless otherwise mentioned. Texts of the reports are expected to be typed in for which you could use software programs like L^AT_EX, Microsoft Word etc. **All screenshots (if any) should be included in this PDF file, clearly titled, and accompanied with adequate descriptions.**
- The PDF file should be clearly formatted to help the TA/instructor to locate the answer to each corresponding question. If the student fails to label the answers with reasonable efforts or submit files with poor readability, the submission may only receive zero or partial score.
- Please use the following naming conventions for your submitted files.
 - Lab2a.asm
 - Lab2b.asm
 - Lab2c.asm
 - Lab2report.pdf
- Students are encouraged to complete each task with best efforts. Even if the final outcome isn't fully correct, partial credit may be awarded if the instructor and TA recognize that the student demonstrates a certain level of understanding.
- Solutions turned in must be your own. Please, mention references (if any) at the end of each question.

Please Complete the Following Task(s).

- **Task a (5 points) Write a MIPS assembly program that prompts the user to enter a string less than 40 characters (no need to test with invalid inputs). The program should then print the entered string back to the console.**

In Lab Report:

- (1) Please take at least two screenshots of the MIPS register panel: one before your program runs, and one after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.
- (2) Please take at least two screenshots of the MIPS memory panel (data tab): one before your program runs, and one after your program finishes.
- (3) Please take at least two screenshots of the console: one before you enter the inputs, and one after your program finishes.

- **Task b (5 points) Write a complete MIPS program that translates and calculates the equation in C shown below.**

```
int main()
{
    int A=15;
    int B=10;
    int C=7;
    int D=2;
    int E=18;
    int F=-3;
    int Z=0;

    Z = (A+B) + (C-D) + (E+F) - (A-C);
}
```

A-F can be stored in temporary registers. **However, the final result Z must be an integer word stored in memory when your program finishes executing. Additionally, when testing the submitted files, we should be able to change the values of A, B, C, D, E, F, and still get the correct results.**

In Lab Report:

(4) Please take at least two screenshots of the MIPS register panel: one before your program runs, and one after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

(5) Please take at least two screenshots of the MIPS memory panel (data tab): one before your program runs, and one after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. In the after-execution capture, **circle the memory location (not register) that contains the final calculated value of Z.**

- **Task c (5 points)** Write a complete MIPS program that implements the same algorithm (in C) shown below.

```
int main()
{
    int A=10;
    int B=15;
    int C=6;
    int Z=0;

    if (A > B || C < 5)
        Z = 1;
    else if ((A > B) && ((C+1) == 7))
        Z = 2;
    else
        Z = 3;

    switch (Z)
    {
        case 1:
            Z = -1;
            break;

        case 2:
            Z = -2;
            break;

        default:
            Z = 0;
            break;
    }
}
```

Note: A-C and Z must be integer words in memory, both when the program begins and when the program ends. In between, they can be stored in registers. Additionally, when testing the submitted files, we should be able to change the values of A, B, C and still get the correct results.

In Lab Report:

(6) Please take at least two screenshots of the MIPS register panel: one before your program runs, and one after your program finishes. Put the register panel in Decimal mode (right-click) so it is easy to see register values.

(7) Please take at least two screenshots of the MIPS memory panel (data tab): one before your program runs, and one after your program finishes. Put the memory panel in Decimal mode (right-click), so it is easy to see memory values. In the after-execution capture, **circle the memory location (not register) that contains the final calculated value of Z.**