Lab 1

Part A (submit a .asm file)

Using MARS simulator to simulate and debug your source code, create a program (using MIPS instruction set) to have the following features:

1. ***Print*** out a message ***string*** with a welcome message "Welcome to Lab 1 exercise! This program will do addition of 2 numbers "
2. ***Print*** out a message ***string*** with the message "Please enter a number between 0-9 "
3. ***Get*** an input from user to enter a ***number*** from 0-9
4. ***Print*** out another message ***string*** with the message "Please enter the next number between 0-9 "
5. ***Get*** an input from user to enter a ***number*** from 0-9
6. Do the process of addition and get the result.
7. ***Print*** out a message ***string*** with the input numbers and result (e.g. "4 + 8 = 12")
8. ***Print*** out another message ***string*** to ask whether the user wants to continue "Do you want to continue? (Y/N)"
9. If the user selects Y, the program loops to step 2. Otherwise, go to the next line.
10. ***End the code***
11. Optimise the code using procedures whenever possible.
    - Use the instructions jal and jr for the function/procedure PrintString as shown below:

```
            jal PrintString
            .....
            .....
PrintString: .....
            .....
            jr $ra
```

12. Put relevant comments. Comments should include:
    - 1 main summary at the start of the program to explain what the program does
    - Comment at main functions/modules
13. Use meaningful variable names/labels.
14. Advanced level (optional):
    - Input validation for steps 3, 5 and 8
15. ***Hints:**
    - **For steps 1-5, 7-8, 10 refer to syscall table**
    - **Make sure the values in the registers are not overwritten if the values are need later, make sure they are transferred to another register or memory if necessary. For example, in step 7, the input values are from step 3 and 5. These values need to be stored somewhere temporarily to be used in step 7.**

Part B (submit a word or pdf file)

Do the following conversions of MIPS instructions and their machine codes. Show your work (steps 1-5) as in the tutorial video:

a. addi $s1, $s2, 0xff11
b. lb $s2,4($s1)
c. 0x012a602a
d. 0x36730098