

**CO200 - Computer Organization and Architecture**  
**A0 - Simulation of RISC-V Assembly using SPIKE**

**Points to Note**

- Write the RISC-V assembly code for all the questions below. Simulate your code on the **spike**.
- In the comments section of each RISC-V assembly code, include a short description of the program, your name, roll number, date of writing the program and other information you deem relevant.
- Your submission is an archive containing the following: One README file + One directory per assignment question. This assignment (A0) has 8 questions (1,...,8) - your submission will be a zip file containing 8 directories and a README file. For each question (in each directory) include the following: RISC-V code, snapshots of the simulator at key execution stages, and other relevant information.
- This is a team assignment. Teams of not more than two. One submission per team.
- (a) Submission is due on September, 3, 9AM. Pack your report, code, screenshots and other files in an archive and mail to [co200.nitk@gmail.com](mailto:co200.nitk@gmail.com).

**Write the RISC-V assembly code for the following.**

1. Load two 64b constants from memory on to x5 and x6. Add them and store the result in x7.
2. Load a 64b constant in the memory location pointed to by the global pointer. Load another 64b constant in the first memory location of the data segment. Add them both and save the sum in the 100<sup>th</sup> doubleword of the data segment.
3. **Hello World Program.** Print “Hello World” on the output screen.
4. Initialize a contiguous chunk of memory to contain 10 two digit decimal numbers. Your program should calculate the sum of these numbers and put the result into x5.
5. Write a program to reverse a string (choose your favorite string as a global parameter).
6. Implement a function that calculates the sum of the elements of the array. The function accepts the length of the array and the address of the first element of the array. The function returns the sum to the caller. The main procedure calculates the average of the elements of the array. The main procedure uses the mentioned the function. Use the RISC-V compiler subroutine conventions for this code.
7. Along the same lines as the previous question, implement a matrix multiplication program using functions.
8. **Factorial Program.** Load a random number in x5. Calculate its factorial using (a) loops, (b) recursion.