

CS527 - Computer Systems Graded Lab Assignment 2

Date: 12th Oct 2020 Deadline: 18th Oct 2020 11.59PM.

Max. marks: 55 points. Weightage: 3.5%.

General Guidelines:

- (i) You need to upload a single zip file containing three files (one for each question, labelled *rollno_q1*, *rollno_q2* and *rollno_q3*). Name the zip file with your roll number.
- (ii) First two lines in each file should be - Your name, Roll no.
- (iii) Include comments (to the point and sufficient) in the code for improved readability. This will also help you to recollect your logic for reuse and during evaluation.
- (iv) Do not use any pseudo instructions.
- (v) Submit before the deadline. Late submissions will be penalized.
- (vi) Plagiarism will be treated seriously.

Question 1: “PALINDROME” (6 points)

Write a procedure to identify whether a given string is a palindrome or not. The string can contain alphanumeric values too.

Marks split: 5 points for logic and 1 point for output.

Question 2: Assembly program to decode a machine code ($8 \times 2.5 + 8 \times 0.5$, i.e., 24 points)

As you are aware, an instruction in RISC-V ISA is encoded and represented as a 32bit (word) machine code. Assume that such a word is stored at a location in memory (say, 0x10001000). Write a program that reads this 32bit word, extracts the *opcode* field to find out if its an instruction or not. If its an instruction, the program should then extract the other fields appropriately and store them in consecutive words beginning at 0x20001000.

To limit the complexity, we'll pick only 8 native RISC-V 32bit instructions, namely, **lw**, **sw**, **add**, **sub**, **xor**, **slli**, **beq**, **jal**. And once an instruction is identified using the fields *opcode*, *func3*, *func6/7*, store the fields in the specified location followed by the values of other fields of the instruction (like, *rs1*, *rs2*, *rd*, *imm*, etc).

Marks split up: 2.5 points for each instruction. 0.5 points for correct output of each instruction.

Question 3: Fibonnaci series (25 Points)

Write a recursive procedure that generates ‘n’ Fibonnaci numbers after the first ‘m’ numbers. Do not consider the first two Fibonnaci numbers (seed values), ‘0’ and ‘1’ in ‘n’ and ‘m’. Give option to change the seed values, ‘n’ and ‘m’ easily. Store the generated ‘n’ fibonnaci numbers at location starting from location 0x10002500.

Marks split: 20 points for recursive logic and 5 points for output.

All the best!