

# ASSIGNMENT-2

IMT2022554: Sreyas Janamanchi

IMT2022555: Pradyumna G

Our submission includes the following files :-

- Sorting.txt: It contains the MIPS code for sorting algorithm, from the first assignment.
- Factorial.txt: It contains the MIPS code to calculate factorial of a value.
- IMT2022554\_IMT2022555\_Non\_pipelined.py: This contains the code for a non-pipelined processor.
- IMT2022554\_IMT2022555Pipelined.py: This contains the code for a pipelined processor.

## IMT2022554 IMT2022555 Non\_pipelined.py

Variables:-

- PC: Stores the value of the instruction in the IF stage.
- Mem: It's the instruction memory that stores all the instructions to be executed.
- DataMem: It is the data memory.
- RegMem: It is the register memory.
- Instructions: Dictionary of instructions as keys with value as the opcode.

Classes:-

- Control\_unit: It replicates the working of a control unit in a processor.

Functions:-

- IF: It takes an instruction as input and returns the opcode. This replicates the functionality of IF stage in a non-pipelined processor
  - ID: It takes the opcode, and control unit as input and returns: rs,rt,rd for R-type instruction; rs,rt,imm for I-type instruction or the address for J-type instruction and updates the control signal according to the instruction that is being executed.
  - EX: It takes the values required to perform the required operations and the control unit and returns “alures”. This replicates the functionality of EX stage in a non-pipelined processor.
  - Memory: It takes the alures, control unit and rt as input and performs the operations to be executed in the memory stage.
  - Writeback: It takes the control unit, alures, data extracted in memory stage and rt as input and writes back to the correct information into the required register if writeback control signal is 1.
- 
- Decimal\_to\_binary: Takes a decimal value and length of binary required as input and returns a binary string.
  - Binary\_to\_decimal: Takes a binary string as input and returns a decimal number as output.

In the main function all the stages are called in order and the returned values are stored to use the values in the next stages.

Sample output for sorting [2,-3,100,10], output array starts from index 24:-

[illegible]

Sample output for calculating 5!, output at index 24:-

[illegible]

IMT2022554 IMT2022555 Pipelined.py

### Variables:-

- The same variables are used, but it also includes the pipelined registers:  
IF\_ID, ID\_EX, EX\_MEM, MEM\_WB and

- Curr\_instructions: A queue that stores the instructions which are being processed.

Classes:-

- Control\_unit
- Instruction: It contains all the methods (IF, ID, EX, MEM, WB), control signals and other necessary information of an instruction.

Functions:-

- All the Instruction methods have been updated to write the return values into their respective pipeline registers.
- In the ID method all the hazard detections are done. These are resolved by forwarding in the appropriate stage. If the forwarding is to be done in a future step, the pipeline\_registers hold flag values, that are used to forward the appropriate information at the appropriate stage.
- When j/beq instructions change the pc. The pipeline registers associated with the previous stages are emptied.

In the main function, curr\_instructions always stores  $\leq 5$  instructions and each instruction moves through curr\_instructions.

Curr\_instructions[n-5]: WB

Curr\_instructions[n-4]: Mem

Curr\_instructions[n-3]: Ex

Curr\_instructions[n-2]: ID

Curr\_instructions[n-1]: IF

In the method of each instruction is called in this schema.

The instruction in the WB is popped after execution.

Sample output for sorting [2,-3,100,10], output array starts from index 20:-

