# Software Version of a Basic Processor

Designed a software version of an in order pipeline processor using Java which works on the RISC set of instructions created by IIT Madras. It consists of the five stages of computing, a single level cache based on the discrete event simulator model.

It consists of five stages Instruction Fetch Stage, Operand Fetch Stage, Execution Stage, Memory Access Stage and the Write Back Stage. And five latches which connect each of the stages and pass on variables created or calculated from one stage to other. This whole processor works on the RISC set of instructions developed by IIT Madras (they even started production of these processors last year). The processor's cache is 1 way Direct-Mapped-Cache. And the memory size is 256KB.The latency of the cache can be adjusted as an argument. The processor consists of 32 registers. The processor saves the instructions into the memory then runs each instruction in a sequential order.

The benchmark consisting of number of instructions processed and the number of cycles taken to process the instructions is specified in it.

The Parsing code which produces the executable file from the code is also included in the folder program for parsing.

A pdf of the set of instructions the processor supports is also provided.

A bunch test cases are also provided which consists of assembly files and the executable files which are produced using the parsing code.

# Inputs and Outputs of the files

For the Parsed Code:

Run the main.java file in the program for parsing folder with arguments `<path-to-assembly-program> <path-to-object-file>`

For the Processor:

Run the main.java file in the processor folder with arguments `<path-to-config-file> <path-to-stat-file> <path-to-object-file>`

This Processor is created as a series of stages/assignments provided in the folder “Stages”.