**Computer Architecture (CSL 216)**
**IIT Jammu, April 2018**
**Assignment 5**

**Problem Statement**: ARM Assembly Program Simulator with multicycle operations

In this assignment you will enhance the instruction set simulator developed in Assignments 3 and 4, to also incorporate pipelining. Make an attempt to be as aggressive as you can with respect to avoiding data and control hazards. As a rule, in such assignments, complete simpler versions of the software before adding complex features. So, to begin with, just stall the pipeline on hazards, and ensure functional correctness.

Print out statistics about the clock cycle counts and average Instructions Per Cycle (IPC) for the input program. Your program should also be capable of displaying, after each instruction, the various instructions that are currently residing in the pipeline, along with the stage that they are in.

The following submissions are due:

1. [**Deadline: 10 April 2018**] **Design document and Test plan**.
    a. Give the algorithm/high-level strategy for simulating the pipeline. This should contain the details of how every instruction type will be handled.
    b. Submit a list of test cases (sample assembly programs) on which your simulator will be tested. Describe the test cases in a document, along with the logic for what feature each test case validates.
2. [**Deadline: 30 April 2018**] **Code and final report.** This should contain:
    a. the algorithm (maybe updated from what was submitted earlier),
    b. test plan (maybe updated), and
    c. any other learnings, experimental observations (e.g., how fast is the simulator in Millions of Instructions simulated Per Second; for what kind of programs is it performing well/poorly?)

**[OPTIONAL: Implement aggressive branch prediction features. You may use your own strategies, or take the help of any research papers. We can give bonus marks if optional features are implemented well, but only if the basic features are complete.]**