25.03.2021

Istanbul Technical Universityn
Department of Computer Engineering

BLG 322E - COMPUTER ARCHITECTURE

Assignment 2

Due Date: Wednesday, April 7, 2021, 23:00.

- Please prepare your homework **using a computer**. Points will be taken off for handwritten submissions.
- Please **write your full name** (first name and last name) **and Student ID** at the top of your solution.
- Submit your solution as a **PDF file** to Ninova before the deadline (**Wednesday,** April 7, 2021, 23:00).
- **No late submissions** will be accepted. Do not send your solutions by e-mail. We will only accept files that have been uploaded to the official Ninova e-learning system before the deadline. Do not risk leaving your submission to the last few minutes.
- **Consequences of plagiarism:** Assignments have to be done individually. Any cheating will be subject to disciplinary action.
- If you have any questions, you may write to the message board or send an e-mail to Research Assistant **Fırat Öncel (oncelf@itu.edu.tr)**.

QUESTION:

Consider the exemplary RISC processor given in Section 2.4.2 of the lecture notes that has an instruction pipeline with the following 5 stages:

- **IF**: Instruction fetch
- **DR**: Instruction Decode, Read registers
- **EX**: Execute
- **ME**: Memory
- **WB**: Write back

Assume the following:

- The CPU does **not** have any **forwarding (bypass)** connections.
- The register file access hazard **is fixed**, i.e., the CPU writes data to registers in the **first** half of the cycle (rising edge) and reads data from registers in the **second** half of the cycle (falling edge).
- The internal structure of the execution circuitry is as shown on slide 2.53 (latest version, 2021), i.e., **branch target address calculation and decision operations** are performed in the **EX** stage, and results are sent directly to the **IF** stage.

```
        SUB     R2, R2, R1
        ADD     R1, $03, R3
        ADD     R1, $01, R2
LOOP:   ADD     R2, $10, R5
        SUB     R3, $01, R3
        ADD     R5, R3, R5
        ADD     R4, $01, R4
        SUB     R2, $01, R2
        BNZ     LOOP
        ADD     R5, $10, R5
        STL     $08(R6), R5
        SUB     R1, R1, R4
        SUB     R1, R1, R3
```

a) We execute the program given on the left in this instruction pipeline. Draw the space-time diagram for the execution of this program. Solve all data and branch conflicts using NOOP instructions. For the given piece of code, what is the total amount of penalty in clock cycles caused by conflicts?

b) To minimize the amount of penalty, apply optimized software-based solutions to the conflicts, if possible. Keep in mind that the results generated by the program cannot be changed. What is the total amount of penalty in clock cycles with the new solutions?

INSTRUCTION SET:

STL	X(Rs), Rm	M[Rs + X] ← Rm	Store
ADD	Ri, Rj, Rd	Rd ← Ri + Rj	Add
SUB	Ri, Rj, Rd	Rd ← Ri - Rj	Subtract
BNZ	Y	PC ← PC + Y	Branch if not zero (Relative)