Hadassah Academic College

Department of Computer Science

Computer Architecture **Exercise 6**

Consider the following program in DLX assembly code:

```
      ADDI R2, R0, #0x180
loop: LW R3, 10(R2)
      SUBI R3, R3, #0x020
      XORI R3, R3, #0x040
      AND R3, R3, #0x0000FFFF
      SW 0(R2), R3
      SUBI R2, R2, #0x4
      BNEZ R2, 0xFFE4
      ADD R3, R8, R9
```

The prefix **0x** indicates hexadecimal representation.

1. Translate each line to formal representation as in the following example:

   ADD R5, R8, R9 **Regs[R5] ← Regs[R8] + Regs[R9]**

2. The program contains a loop that begins with the label **loop** and continues to the conditional branch. Explain briefly how the offset **0xFFE4** in the branch instruction causes the program to return to the start of the loop.

3. Find the number of loop iterations in running the program.

4. Find the number of instructions that run in the program (reminder: an instruction that runs N times over N loop iterations counts as N instructions).

5. Find the number of instructions of each instruction type that run during the program.
   Find the relative distribution IC_i/IC for the 4 instruction types

6. Find the CPI for the program on the DLX using the CPU equation.