COMP 222 Computer Organization
Assignment #5—Instruction-Level Parallelism

Objective:

To calculate the performance of a program with dependent arithmetic instructions, by simulating the execution on (i) a simple 6-stage pipeline architecture; (ii) a superscalar 6-stage architecture of degree 2 (2 sets of functional units). The six stages are: Fetch Instruction (FI), Decode Instruction (DI), Calculate Operands (CO), Fetch Operands (FO), Execute Instructions (EI), and Write Output (WO).

Inputs:

- Number of instructions in the program
- Set of automatically numbered instructions containing arithmetic register assignments
- Architecture to simulate program on (pipelined, superscalar)

Outputs:

- The total cycle count for the program
- A Gantt chart of the pipelined stages of the instructions

Specification:

The program calculates the performance of a set of arithmetic register assignment statements and prints out the aligned pipelined instructions based on choosing from a menu of choices, where each choice calls the appropriate procedure:

a) Enter instructions
b) Calculate total cycle count on a 6-stage pipelined architecture
c) Calculate total cycle count on a 6-stage superscalar architecture
d) Quit program

{**Hint:** to recognize data dependencies, store the register indices in a **struct** containing fields for the destination register index, both source register indices, and the dependency delay. A data dependency occurs when the destination register of instruction (i) is the same as at least one of the source registers of instruction (i+k). The delay is calculated based on the distance between instruction (i) and instruction (i+k), i.e. the value of k }

(**Printing hint:** To align stages for readability, use printf("\t") to tab the stages)

What to turn in:

- The source code as a single C file uploaded to Canvas (http://canvas.csun.edu) by the deadline (-20% per consecutive day for late submissions, up to the 4th day).

```
% ./a.out

Instruction-level parallelism
------------------------------
a) Enter instructions
b) Calculate total cycle count on a 6-stage pipelined architecture
c) Calculate total cycle count on a 6-stage superscalar architecture
d) Quit program

Enter selection: a

Enter total number of instructions: 5

1) r0=r1+r2
2) r1=r0+r3
3) r2=r3+r2
4) r0=r2+r0
5) r3=r3+r3

Instruction-level parallelism
------------------------------
a) Enter instructions
b) Calculate total cycle count on a 6-stage pipelined architecture
c) Calculate total cycle count on a 6-stage superscalar architecture
d) Quit program

Enter selection: b

Total number of cycles: 14

1)FI    DI      CO      FO      EI      WO
2)                      FI      DI      CO      FO      EI      WO
3)                              FI      DI      CO      FO      EI      WO
4)                                              FI      DI      CO      FO      EI      WO
5)                                                      FI      DI      CO      FO      EI      WO

Instruction-level parallelism
------------------------------
a) Enter instructions
b) Calculate total cycle count on a 6-stage pipelined architecture
c) Calculate total cycle count on a 6-stage superscalar architecture
d) Quit program

Enter selection: c

Total number of cycles: 12

1)FI    DI      CO      FO      EI      WO
2)                      FI      DI      CO      FO      EI      WO
3)                      FI      DI      CO      FO      EI      WO
4)                                              FI      DI      CO      FO      EI      WO
5)                                              FI      DI      CO      FO      EI      WO


Instruction-level parallelism
------------------------------
a) Enter instructions
b) Calculate total cycle count on a 6-stage pipelined architecture
c) Calculate total cycle count on a 6-stage superscalar architecture
d) Quit program

Enter selection: d
```