Surname	
Name	
POLIMI Personal code	
Signature	

Politecnico di Milano, 6 May 2024

Course on Advanced Computer Architectures

Prof. C. Silvano

EX 1	**(4 points)**	
EX 2	**(5 points)**	
EX 3	**(4 points)**	
QUIZ 4	**(1 point)**	
QUIZ 5	**(1 point)**	
TOTAL	**(15 points)**	
QUIZ 6 OPTIONAL	**+ 3 extra points**	

EXERCISE 1: VLIW (4 points)

Let's consider the following assembly code:

```
INIT:  ADDUI $R1, $R0, 0
       ADDUI $R2, $R0, 40
       ADDUI $R4, $R0, 8

LOOP1: LD $F0, 0 ($R1)
       FADD $F4, $F0, $F2
       SD $F4, 0 ($R1)
       ADDUI $R3, $R0, 0

LOOP2: LD $F6, 0($R3)
       FADD $F8, $F6, $F2
       SD $F8, 0($R3)
       ADDUI $R3, $R3, 4
       BNE $R3, $R4, LOOP2

       ADDUI $R1, $R1, 4
       BNE $R1, $R2, LOOP1
```

1. *Complete the code of the first iteration of the outer loop* ***LOOP1:***

```
LOOP1: LD $F0, 0 ($R1)
       FADD $F4, $F0, $F2
       SD $F4, 0 ($R1)
       ADDUI $R3, $R0, 0
```

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

2. *Now schedule **the first iteration of the outer loop LOOP1** by using **the list-based scheduling** (do NOT introduce any software pipelining, loop unrolling and modifications to loop indexes) on a **3-issue VLIW** machine with **fully pipelined functional units:***

   - ***1 Memory Unit with 2 cycle latency***
   - ***1 Integer ALU with 1 cycle latency to next Int/FP/LD/SD & 2 cycle latency to next Branch***
   - ***1 FP ALU with 2 cycle latency***

   ***No branch prediction.** The branch is completed with **1 cycle delay slot** (branch solved in ID stage). In the Register File, it is possible to read and write at the same address at the same clock cycle. Please do not write in NOPs.*

	Memory Unit	Integer Unit	Floating Point Unit
C1	`LD $F0, 0 ($R1)`		
C2			
C3			
C4			
C5			
C6			
C7			
C8			
C9			
C10			
C11			
C12			
C13			
C14			
C15			
C16			
C17			
C18			
C19			
C20			
C21			
C22			
C23			
C24			
C25			

3. How long is the critical path? _____

4. What performance did you achieve in CPIas?

   _____

5. What performance did you achieve in FP ops per cycles?

   _____

6. How much is the code efficiency?

   _____

EXERCISE 2: DYNAMIC BRANCH PREDICTION (5 points)

Let's consider the same assembly code used for **EXERCISE 1:**

```
INIT:  ADDUI $R1, $R0, 0
       ADDUI $R2, $R0, 40
       ADDUI $R4, $R0, 8

LOOP1: LD $F0, 0 ($R1)
       FADD $F4, $F0, $F2
       SD $F4, 0 ($R1)
       ADDUI $R3, $R0, 0

LOOP2: LD $F6, 0($R3)
       FADD $F8, $F6, $F2
       SD $F8, 0($R3)
       ADDUI $R3, $R3, 4
       BNE $R3, $R4, LOOP2

       ADDUI $R1, $R1, 4
       BNE $R1, $R2, LOOP1
```

1. *How many iterations for the outer loop **LOOP1?***

   _____

2. *How many iterations for the inner loop **LOOP2?***

   _____

3. *How many branch instructions are executed in the code?*

   _____

4. Assuming there is **no branch prediction** and each branch costs **2 cycle penalty** to fetch the correct instruction, *how many branch penalty cycles are needed to execute both loops?*

   _____

5. Assuming to execute the code on a pipelined processor with a dynamic **Branch Prediction Unit (BPU)** in the **IF-stage** composed of:
   - **2-entry 2-bit Branch History Table**
   - **2-entry Branch Target Buffer**

Let's assume the 2 branch instructions **do not collide** so they are allocated to the 2 entries of the BPU where the **BTB hit**, there are 4 cases for each conditional branch with the related **branch penalty cycles:**

	Branch Outcome	
	Taken	**Not Taken**
Strongly Taken	**1 cycle**	**2 cycles**
Weakly Taken	**1 cycle**	**2 cycles**
Strongly Not Taken	**2 cycles**	**0**
Weakly Not Taken	**2 cycles**	**0**

Let's assume the 2-entries do not collide with BTB hit and are initialized as **Strongly Taken**, *please complete the following table:*

Explain the branch behavior considering the inner LOOP2 in isolation.	***How many branch penalty cycles to execute the LOOP2 in isolation?***	***Calculate the branch misprediction rate to execute the LOOP2 in isolation.***
Explain the branch behavior considering both loops.	***How many branch penalty cycles to execute both loops?***	***Calculate the global branch misprediction rate to execute both loops.***

EXERCISE 3 – SCOREBOARD (4 points)

1. *Let's consider the following assembly code containing multiple types of dependences. Complete the following table by inserting all types of data-dependences, anti-dependences and output dependences for each instruction:*

INSTRUCTION	ANALYSIS OF DEPENDECES
`I0: LD $F2,A($R6)`	--
`I1: FADD $F3,$F2,$F6`	True data dependence with I0 for $F2
`I2: SD $F3,A($R7)`	
`I3: LD $F3,B($R6)`	
`I4: SD $F3,C($R7)`	
`I5: ADDUI $R6,$R6,4`	
`I6: ADDUI $R7,$R7,4`	

2. *Schedule the code on a CPU with dynamic scheduling based on **OPTIMIZED SCOREBOARD** with the following assumptions:*
   - 2 LOAD/STORE Units (LDU1, LDU2) with latency 3 cycles
   - 1 FP Unit (FPU1) with latency 3 cycles
   - 1 ALU/BR Unit ALU1 with latency 1 cycle
   - Register Filw with 2 read ports and 1 write port
   - Check for WAR and WAW hazards postponed to the WRITE BACK phase
   - Forwarding

INSTRUCTION	ISSUE	READ OPs	EXEC COMPL.	WRITE BACK	Hazards Type Forwarding	UNIT
`I0: LD $F2,A($R6)`	1	2	5	6		LDU1
`I1: FADD $F3,$F2,$F6`	2	6	9	10	RAW $f2 by forw.	FPU1
`I2: SD $F3,A($R7)`						
`I3: LD $F3,B($R6)`						
`I4: SD $F3,C($R7)`						
`I5: ADDUI $R6,$R6,4`						
`I6: ADDUI $R7,$R7,4`						

3. *Express the formula and calculate the CPI:*
   CPI = _____

QUIZ 4 – SPECULATIVE TOMASULO (1 point)

In the speculative Tomasulo architecture, exceptions are taken when the instruction that generated them reaches the head of the ROB.

(TRUE/FALSE ANSWER)
1 point

Answer: **TRUE** **FALSE**

Motivate your answer:

_____

_____

QUIZ 5 – CACHE PERFORMANCE (1 point)

Let us consider a computer architecture with L1 and L2 caches with the following parameters:

- Processor Clock Frequency = 1 GHz
- Hit Time L1 = 1 clock cycle
- Hit Rate L1 = 95%
- Hit Time L2 = 5 clock cycles
- Hit Rate L2 = 90%
- Miss Penalty L2 = 15 clock cycles

How much is the Global Miss Rate for Last Level Cache?
(SINGLE ANSWER)
1 point

Answer 1: 5% **T**

Answer 2: 0.5% **T**

Answer 3: 10% **T**

Answer 4: 7.5% **T**

Answer 5: 25% **T**

Motivate your answer:

_____

_____

_____

QUIZ 6: CACHE MEMORIES (3 points) OPTIONAL

Given a cache of a given capacity, associativity and block size, answer **TRUE** or **FALSE** to the following questions, ***motivating your answers***.

- Doubling the cache capacity of a direct mapped cache usually reduces conflict misses

  Answer: **TRUE** **FALSE**

_____

_____

_____

_____

_____

- Doubling the block size reduces compulsory misses **TRUE** **FALSE**

_____

_____

_____

_____

_____

- Change the nesting of loops in the code to access data in order stored in memory will increase spatial locality, possibly reducing the miss rate **TRUE** **FALSE**

_____

_____

_____

_____

_____