Surname (COGNOME)	
Name	
POLIMI Personal Code	
Signature	

Politecnico di Milano, 11 July, 2023

Course on Advanced Computer Architectures

Prof. C. Silvano

EX1	**(5 points)**	
EX2	**(5 points)**	
EX3	**(5 points)**	
Q1	**(5 points)**	
Q2	**(5 points)**	
QUIZZES	**(8 points)**	
TOTAL	**(33 points)**	

EXERCISE 1 – TOMASULO (5 points)

Let's consider the following assembly code to be executed on a CPU with dynamic scheduling based on **TOMASULO algorithm** with all cache HITS, a single Common Data Bus and:

- *2 RESERVATION STATIONS **(RS1, RS2)** with 2 LOAD/STORE units **(LDU1, LDU2)** with latency **4***
- *2 RESERVATION STATIONS **(RS3, RS4)** with 2 INTEGER ALU/BR units **(ALU1, ALU2)** with latency **2***
- *2 RESERVATION STATIONS **(RS5, RS6)** with 2 FP ALU units **(FPU, FPU2)** with latency **4***

Please complete the following table:

INSTRUCTION	ISSUE	START EXEC	WRITE RESULT	Hazards Type	RSi	UNIT
`I1: lw $f1,A($r1)`	1	2	6	--	RS1	LDU1
`I2: addi $r2,$r1,4`						
`I3: ld $f2,A($r2)`						
`I4: addi $r3,$r1,8`						
`I5: ld $f3,A($r3)`						
`I6: fadd $f1,$f1,$f2`						
`I7: fadd $f2,$f2,$f3`						
`I8: fmul $f1,$f1,$f2`						

Calculate the CPI:

CPI = ______________________________

EXERCISE 2 – VLIW (5 points)

Let's consider the following LOOP code:

```
LOOP:LD F1, A(R1)
     ADDUI R2, R1,4
     LD F2, A(R2)
     ADDUI R3, R1,8
     LD F3, A(R3)
     FADD F1, F1, F2
     FADD F2, F2, F3
     FMUL F1, F1, F2
     SD F1, B(R1)
     ADDUI R1, R1,4
     BNE R1, R2, LOOP
```

Given a **4-issue VLIW** machine with **fully pipelined functional units:**

- **1 Memory Units with 3 cycles latency**
- **2 FP ALUs with 2 cycles latency**
- **1 Integer ALU with 1 cycle latency to next Int/FP & 2 cycle latency to next Branch**

The branch is completed with 1 cycle delay slot (branch solved in ID stage). **No branch prediction.**
In the Register File, it is possible to read and write at the same address at the same clock cycle.
Considering one iteration of the loop, complete the following table by using the ***list-based scheduling*** (do NOT introduce any software pipelining, loop unrolling and modifications to loop indexes) on the *4-issue VLIW machine including the BRANCH DELAY SLOT.* Please do not write in NOPs.

	Memory Unit 1	Floating Point Unit 1	Floating Point Unit 2	Integer Unit
C1	LD F1, A(R1)			
C2				
C3				
C4				
C5				
C6				
C7				
C8				
C9				
C10				
C11				
C12				
C13				
C14				
C15				

1. How long is the critical path? ________________________________

2. What performance did you achieve in CPIas?

   ____

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

   ____

4. How much is the code efficiency?

   ____

5. Assuming to have **1 FP ALU and 2 INTEGER UNITs** how much is the impact on CPI and code efficiency?

____

____

EXERCISE 3 – MESI PROTOCOL (5 points)

Let’s consider the following access patterns on a dual processor system with a direct-mapped, write-back cache with one cache block per processor and a 2 cache block memory. Assume the **MESI protocol** is used, with **write-back** caches, **write-allocate,** and **write-invalidate** of other caches.

Please complete the following table:

Cycle	**After Operation**	**P0 cache block state**	**P1 cache block state**	**Memory at block 0 up to date?**	**Memory at block 1 up to date?**
1	**P1: read block 0**	Exclusive (1)	Exclusive (0)	Yes	Yes
2		Modified (1)	Exclusive (0)	Yes	No
3		Modified (1)	Exclusive (0)	Yes	No
4		Modified (1)	Exclusive (0)	Yes	No
5		Modified (0)	Invalid	No	Yes
6		Modified (0)	Modified (1)	No	No
7		Shared (0)	Shared (0)	Yes	Yes
8		Exclusive (1)	Exclusive (0)	Yes	Yes

QUESTION 1: CACHE MEMORIES (5 points)

Describe the main impacts of the 3 different techniques used to design cache memories on the 3 different types of cache misses:

	Compulsory misses	***Capacity misses***	***Conflict misses***
Direct Mapped Cache			
Set Associative Cache			
Fully Associative Cache			

QUESTION 2: DIRECTORY-BASED PROTOCOL (5 points)

Let's consider a directory-based protocol for a distributed shared memory system with 4 Nodes (N0, N1, N2, N3) where: **Directory N0 Block B0 | State: Uncached | Sharer Bits: - - - - |**

Given the following sequence:
Read Miss on B0 from node N1;
Read Miss on B0 from node N2;
Write Hit on B0 from node N1;

Please answer to the following questions:

After the two Read Misses, what is the status of the block B0 in the directory N0?	**Directory N0 Block B0** **| State: __________ | Sharer Bits: ________ |**
What are the home node, the local node and the remote node(s) during the Write Hit?	
What is the sequence of messages sent between the nodes during the Write Hit?	
Which is the state of the block B0 in the local cache and in the remote cache at the end of the sequence?	
Which is the state of the block B0 in the home directory at the end of the sequence?	**Directory N0 Block B0** **| State: __________ | Sharer Bits: ________ |**

MULTIPLE-CHOICE QUESTIONS: (8 points)

Question 1 (format Multiple Choice – Single answer)

Let's consider a fully associative write-back cache with many cache entries that at cold start is empty and receives the following sequence of memory accesses:

Read Mem[AAAA]
Write Mem[AAAA]
Read Mem[BBBB]
Write Mem[BBBB]
Read Mem[AAAA]
Read Mem[BBBB]

When using a ***"write allocate"*** versus a ***"no-write allocate"*** policy, how many cache hits and misses are there?
(SINGLE ANSWER)
2 points

Answer 1: Write allocate has 4 hits & 2 misses | No-write allocate has 2 hits & 4 misses
Answer 2: Write allocate has 4 hits & 2 misses | No-write allocate has 4 hits & 2 misses
Answer 3: Write allocate has 5 hits & 1 miss | No-write allocate has 5 hits & 1 miss
Answer 4: Write allocate has 3 hits & 3 misses | No-write allocate has 2 hits & 4 misses
Answer 5: Write allocate has 2 hits & 4 misses | No-write allocate has 4 hits & 2 misses

Question 2 (format Multiple Choice – Single answer)
Let's consider a directory-based protocol for a distributed shared memory system with 4 Nodes (N0, N1, N2, N3) where we consider the block B1 in the directory of N1:
Directory N1 Block B1 | State: Shared | Sharer Bits: 0011 |

*During a **Write Miss** on B1 from N0, please indicate which are the home node, the local node and the remote node(s):*
(SINGLE ANSWER)
1 point

Answer 1: N1 home node, N2 local node; N3 remote node.
Answer 2: N1 home node; N0 local node; N2 remote node.
Answer 3: N0 home node; N1 local node; N2 and N3 remote nodes.
Answer 4: N1 home node; N0 local node; N2 and N3 remote nodes.
Answer 5: N1 home node; N1 local node; N3 remote node.
Question 3 (format Multiple Choice – Multiple answers)

What characteristics make a program suitable to be accelerated on a GPU?
(MULTIPLE ANSWERS)
1 point

Answer 1: Extensive data parallelism;
Answer 2: Many if-else constructs;
Answer 3: Frequent communication between parallel tasks;
Answer 4: Few synchronization points;
Answer 5: Many mathematical operations on each data;

Question 4 (format Multiple Choice – Single answer)

Let's consider the following code executed by a Vector Processor with:

- *Vector Register File composed of 32 vectors of 8 elements per 64 bits/element;*
- *Scalar FP Register File composed of 32 registers of 64 bits;*
- *One Load/Store Vector Unit with operation chaining and memory bandwidth 64 bits;*
- *One ADD/SUB Vector Unit with operation chaining.*
- *One MUL/DIV Vector Unit with operation chaining.*

L.V V1, RX	# Load vector from memory address RX into V1
L.V V3, RY	# Load vector from memory address RY into V2
MULVS.D V1, V1, F0	# FP multiply vector V1 to scalar F0
ADDVV.D V2, V1, V1	# FP add vectors V1 and V1
MULVS.D V2, V2, F0	# FP multiply vector V2 to scalar F0
ADDVV.D V3, V2, V3	# FP add vectors V2 and V3
S.V V3, RZ	# Store vector V3 into memory address RZ

How many convoys? How many clock cycles to execute the code?

(SINGLE ANSWER)
2 points

Answer 1: 3 convoys; 24 clock cycles
Answer 2: 4 convoys; 32 clock cycles
Answer 3: 5 convoys; 40 clock cycles

Question 5 (format Multiple Choice – Multiple answers)

What are the main **disadvantages** of the static scheduling used to support ILP?

(MULTIPLE ANSWER)
1 point

Answer 1: Variable memory latency due to unpredictable cache misses
Answer 2: Large amount of parallelism available within a basic block
Answer 3: The need of a cache coherency protocol
Answer 4: Code size increase due to the insertion of NOPs
Answer 5: Runtime detection and resolution of data dependencies
Answer 6: Increment of hardware complexity and power dissipation

Question 6 (format Multiple Choice – Single answer)

The reservation stations provided by Tomasulo are effective to:

(SINGLE ANSWER)
1 point

Answer1: *Avoid WAR and WAW hazards by implicit register renaming*

Answer2: *Shorten RAW hazards by providing the results directly from them*

Answer3: *Both of them*