Computer Architectures
Exam of 22.2.2022 – part I – B

First name, Last name, ID.....

Question #1

You are requested to

1. Explain what *Loop Unrolling* is, stating who is in charge of applying it
2. Describe the advantages and disadvantages it introduces
3. Report the code resulting from the application of Loop Unrolling to the following code:

```
for (i=0;i<MAX;i++ )
{
    y[i] = x[i]+ 5;
}.
```

1. Loop unrolling is a static technique based on reducing the number of iterations a given loop is executed, modifying its body. It is normally implemented by the compiler.
2. Loop unrolling improves the performance by
    a. Reducing the number of branches
    b. Increasing the size of the loop body, thus increasing the chances of identifying ILP in it

   Its main disadvantage lies in the increased code size.

3.

```
y[0] = x[0]+ 5;
y[1] = x[i+1]+ 5;
y[2] = x[i+2]+ 5;
y[3] = x[i+3]+ 5;
…
y[MAX-1] = x[MAX-1]+ 5;
```

Computer Architectures
Exam of 22.2.2022 – part I – B

First name, Last name, ID.....

Question #2

Let consider a MIPS64 architecture including the following functional units (for each unit the number of clock periods to complete one instruction is reported):

- Integer ALU: 1 clock period
- Data memory: 1 clock period
- FP arithmetic unit: 2 clock periods (pipelined)
- FP multiplier unit: 6 clock periods (pipelined)
- FP divider unit: 8 clock periods (unpipelined)

You should also assume that

- The branch delay slot corresponds to 1 clock cycle, and the branch delay slot is not enabled
- Data forwarding is enabled
- The EXE phase can be completed out-of-order.

You should consider the following code fragment and, filling the following tables, determine the pipeline behavior in each clock period, as well as the total number of clock periods required to execute the fragment. The value of the constant k is written in f10 before the beginning of the code fragment.

```
; ********************* MIPS64 ***********************
;  for (i = 0; i < 10; i++) {
;       v4[i] = (v1[i] + v2[i]*k)/v3[i];

;  }
```

```
                .data
v1:     .double "10 values"
v2:     .double "10 values"
v3:     .double "10 values"
v4:     .double "10 values"

                .text
main:  daddui r1,r0,0  I1
       daddui r2,r0,10  I2
loop:  l.d  f1,v1(r1)  I3
       l.d  f2,v2(r1)  I4
       mul.d  f6, f2, f10  I5
       l.d  f3,v3(r1)  I6
       add.d  f7, f1, f6  I7
       div.d f8, f7, f3  I8
       s.d  f8,v4(r1)  I9
       daddui  r1,r1,8  I10
       daddi  r2,r2,-1  I11
       bnez  r2,loop  I12
       halt  I13
```

	Comments	Clock cycles
	r1← pointer	5
	r2 <= 10	1
	f1 <= v1[i]	1
	f2 <= v2[i]	1
	f6 <= v2[i]*k	7
	f3 <= v3[i]	0
	f7 <= v1[i]+v2[i]*k	2
	f8 <= (v1[i]+v2[i]*k) / v3[i]	8
	v4[i] <= f8	1
	r1 <= r1 + 8	1
	r2 <= r2 - 1	1
		2
		1
total		256

Computer Architectures
Exam of 12.5.2022 - part I

First name, Last name, ID...

main: daddui r1,r0,0	F	D	E	M	W																																5
daddui r2,r0,10		F	D	E	M	W																															1
loop: l.d f1,v1(r1)			F	D	E	M	W																														1
l.d f2,v2(r1)				F	D	E	M	W																													1
mul.d f6, f2, f10					F	D		E	E	E	E	E	E	M	W																						7
l.d f3,v3(r1)						F		D	E	M	W																										0
add.d f7, f1, f6								F	D					E	E	M	W																				2
div.d f8, f7, f3									F					D		E	E	E	E	E	E	E	E	M	W												8
s.d f8,v4(r1)														F		D	E								M	W											1
daddui r1,r1,8																F	D								E	M	W										1
daddi r2,r2,-1																	F								D	E	M	W									1
bnez r2,loop																									F		D	E	M	W							2
halt																											F	D	E	M	W						1