DUE IN CLASS

STUDENTS IDENTIFICATION:

Number:	Name:
102908	Luana Ferraz
103 091	Pedro Cuito
103095	Paniel Nuncs

2.1 Simple execution, without data forwarding techniques

e)	Clock cycles	18		Instructions	-	7		Average CPI	2.57-1	
----	--------------	----	--	--------------	---	---	--	-------------	--------	--

f)	Clock cycles	174	Stalls
	Instructions	61	
	Average CPI	2.852	

Stalls: - Data	101
- Structural	o
- Branch Taken	\mathcal{B}

Ma exercico do programa, pede-se observar que ocorrem 8 branch Taken Stalls, num total de 9 ciclos exerutados; apris cala instrução "bne", a instrução sequinte comega a sex exerutada más não termina; lago, é flushed uma viz que a branch prediction foi incorreta. Lago, a branch prediction policy adotada pelo simulador ó, evitado Predict pot taken, o que provoca com que ocorram consistentemente stalls do pipeline uma viz que o branch ó, na vixado etaken.

2.2 Application of data forwarding techniques

c)	Clock cycles	136
	Instructions	61
	Average CPI	2 130

Stalls: - Data	63
- Structural	9
- Branch Taken	R

d) Greedup =
$$\frac{t \text{ antigo}}{t \text{ novo}} = \frac{\text{# cycles antigo} \times t \text{ cycle}}{\text{# cycles novo} \times t \text{ cycle}} = \frac{174}{136} \approx 1.279$$

2.3 Source code optimization: minimization of data and structural hazards

a) Attach a copy of the new assembly program.

c)	Clock cycles	118
	Instructions	61
	Average CPI	a a 34

Stalls: - Data	36
- Structural	q
- Branch Taken	, Q

d)

Apreloup =
$$\frac{t \text{ antiso}}{t \text{ novo}} = \frac{\text{# cycles antiso} \times t \text{ cycle}}{\text{# cycles novo} \times t \text{ cycle}} = \frac{174}{118} \approx 1.475$$

2.4 Source code optimization: loop unrolling

a) Attach a copy of the new assembly program.

c)	Clock cycles	91
	Instructions	49
	Average CPI	1.957

Stalls:	- Data	51
	- Structural	9
	- Branch Taken	2

d)

4 peedup =
$$\frac{t \text{ antigo}}{t \text{ novo}} = \frac{\text{# cycles antigo} \times t \text{ cycle}}{\text{# cycles novo} \times t \text{ cycle}} = \frac{174}{91} \sim 1.91 \text{ d}$$

2.5 Source code optimization: branch delay slot

a) Attach a copy of the new assembly program.

d)	Clock cycles	101
	Instructions	61
	Average CPI	1.656

Stalls: - Data	27
- Structural	q
- Branch Taken	0

e) Greedup =
$$\frac{t \text{ antigo}}{t \text{ novo}} = \frac{\text{# cycles antigo} \times \text{tcycle}}{\text{# cycles novo} \times \text{tcycle}} = \frac{174}{101} \approx 1.723$$

Table 1: Pipeline time diagram, with data forwarding techniques.

ı				Ι_	Ι.	1_		Ι_	Ι.	Ι.		Ι	I	l	l			T									I					1	I	T	T						
\vdash	INSTRUCTIONS	1	_	_	_	5		7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38 .	39	40
1	lw \$12, 0(\$1)	F	V	χ	M	W																												₩					4	\dashv	<u> </u>
2	dmul \$12, \$12, \$9		F	7	Mo	R	M	Ma	Mz	Mu	Ms	146	M	W																				lacksquare					ightharpoonup	_	<u> </u>
3	dadd \$ 9, \$ 9, \$12			F	0	R	χ	R	R	R	R	A	5	M	Ψ																			<u> </u>							
4	dodi \$ 5, \$ 5, 1				F	F	D	P	D	\mathcal{D}	V	D	V	χ	M	W																									
5	daddi \$ 1, \$ 1,8						9 F	F	F	F	F	F	F	0	χ	M	W																								
6	bne \$ 6, \$ 5, loop													F	D	χ	Μ	W																							
7	5 w \$ 9, mult (\$0)														۴																										
8	lw \$12, 0 (\$1)															F																									
9	,																																								
10																																									
11																																									
12																																									
13																																									
14																																									
15																																									
16																																									
17																																									
18																																									
19																																									
20																																									
21																																									
22																																									
23																																									
24																																									
25																																									
26																																									
27																																									
28																																							T		
29																																							T		
30																																							T		

Table 2: Pipeline time diagram, with minimization techniques to reduce the data and structural hazards.

ı				1	1	1		1		1	1	1		1		1	1		1						ı		-	-	-	- 1				1				_			_
	INSTRUCTIONS	1	4	_	_	5	_	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37 3	38 3	39	10
1	(w \$12, 0/\$1)	F	9	χ	M	W																																	_	_	
2	dadd: \$5,\$5,1		F	D	χ	M	W																																		
3	dwul \$12.\$12.\$9			F	0	H ₀	M ₁	Mz	Hz	My	Ms	Ma	M	W																											
4	daddi \$1,51,8				۶	Д	Χ	М	W																																
5	dadd \$ 9.59,512					F	1	X	R	R	R	R	9	M	W																										
6	dodd: \$1,\$1,8 dodd \$9,\$9,\$12 bne \$6,\$5,100p						F	D	D	\mathcal{O}	D	D	D	χ	M	W																									
7	sw \$ 9, mult (\$0)							F	۶	۴	F	F	F																												
8	SW \$ 9, mult (\$0) [W \$ 12,0 (\$1)													F																											
9	, ,																																								
10																																									
11																																									
12																																									
13																																									
14																																									
15																																									
16																																									
17																																									
18																																									
19																																									
20																																									
21																																									
22																																						T	T	T	
23																																						T	T	寸	٦
24																																						T	T	\exists	\exists
25																																						T	T	T	
26																																						T	T	T	
27																																						T	T	寸	٦
28																																						T	寸	\forall	\exists
29																																						7	\forall	寸	
30																																						T	\top	\exists	٦
ئت			<u> </u>	<u> </u>	1	1	1	<u> </u>	1	<u> </u>		<u> </u>	1	<u> </u>	1	<u> </u>		l	l					<u> </u>						i									ㅗ		

Table 3: Pipeline time diagram: usage of loop unrolling minimization techniques to reduce the control hazards.

	INS	STRUCTIONS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	2 33	34	35	36	37	38	39	40
1	dmi	\$ 16,\$12,\$9 \$ 16,\$12,\$9 \$ 12,16 (\$1) \$ 2,\$2,\$16 \$ 17,\$13,\$9 \$ 1,\$1,\$6 \$ 13,\$ (\$1) \$ 9,\$9,\$17 \$ 18,\$14,\$9 \$ 5,\$5,1 \$ 14,46 (\$1) \$ 9,\$2,\$18 \$ 9,\$5,600 \$ 1,\$5,600 \$ 1,\$5,600 \$ 1,\$5,600 \$ 1,\$5,600 \$ 1,\$5,600 \$ 1,\$00 \$ 1,\$0	F	0	H,	MA	М .	Mz	Mu	Ms	146	M	W																													
2	7099	\$1,\$1,8		F	1	χ	М	W		_																																
3	lw	\$ 12,16 (\$1)			F	9	χ	Μ	W																																	
4	dad	3 9 13 9 16				F	D	χ	R	R	a	9	М	W																												
5	g mu	\$17, \$13, 59					۴	0	A	A	R	R	Ma	M.	Ma	M3	μu	Ms	M6	M	W																					
6	dadd	\$1,31.16						۲	F	F	F	F	1	X	М	W																										
7	1W	\$13,8(\$1)											F	D	χ	M	3																									
8	Popp	39, 89, 817												F	1)	X	A	R	ø	9	M	W																				
9	amul	318, 314, 39													F	D	R	A	A	R	M,	M ₁	Ŋο	M3	Mu	Ms	M6	М	W													
10	дадд	\$5, \$5,1														F	۴	F	Ŧ	F	1	χ	M	W																		
11	W	\$14,16(91)																			F	V	χ	M	W											L						
12	9999	39,59,518																				F	0	χ	a	A	A	5	N	W						L						
13	bne	36, 35, 1000																					F	0	V	V	D	P	χ	M	W					L						
14	gw/	\$9, mult (30)																						F	F	F	F	F								L			Ш			
15		,																																	igg	L	<u> </u>		Ш			
16																																				L	<u> </u>		Ш			
17																																				L	<u> </u>		Ш			
18																																				L	<u> </u>		Ш			
19																																				L	<u> </u>		Ш			
20																																			lacksquare	Ļ	$oldsymbol{ol}}}}}}}}}}}}}}}}}}$	\bigsqcup	Ш			
21																																			lacksquare	Ļ	$oldsymbol{ol}}}}}}}}}}}}}}}}}}$	\bigsqcup	Ш			
22																																			lacksquare	Ļ	$oldsymbol{ol}}}}}}}}}}}}}}}}}}$	\bigsqcup	Ш			
23																																			\downarrow	\downarrow	\perp	\bigsqcup	\sqcup	_		
24																																			\downarrow	\downarrow	\perp		\sqcup	_	\dashv	
25																																			\downarrow	\perp	\perp		\sqcup	\blacksquare	\dashv	
26																																			$oldsymbol{\perp}$	\downarrow	ot	\bigsqcup	Ш	ightharpoonup	_	
27																																			$oldsymbol{\perp}$	\downarrow	ot	\bigsqcup	Ш	ightharpoonup	_	
28																																			$oldsymbol{\perp}$	\downarrow	ot	\bigsqcup	Ш	ightharpoonup	_	
29																																			\downarrow	\downarrow	\perp	\bigsqcup	Щ	_		
30																																										

Table 4: Pipeline time diagram: usage of branch delay slot techniques to reduce the control hazards.

[INSTRUCTIONS	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	7 18	19	20	21	1 22	2 23	3 24	25	26	27	28	29	30	31	32	2 33	34	35	36	37	38	39	40
1	lw \$12,0(\$1) dodd: \$5,55,1 dmul \$12,\$12,\$9 hodd: \$7,\$1,\$1,8 bne \$6,\$5,60p dadd \$9,\$12,\$12	F	1	D	0	Q F	n	X	M	W																							T		1			\Box			
2	2021 65 651		F	F	F	F	F	ก	χ	Μ	W																								T	T		П	\exists	T	
3	dunul & 12, 512,39							F	P	Mo	MJ	112	Mz	Mu	Ms	M6	W	W																							
4	20dd; \$1, \$1.8								F	0	Χ	М	W																												
5	bue \$ 6, \$5, loop									8	D	χ	Μ	W																											
6	add 99, 59,512										F	0	χ	R	R	R	5	Μ	W																						
7	(W \$ 12,0 (\$1)											F																							L	L		Ш			
8																																			L	L		Ш			
9																																			L	L		Щ	_		_
10																																	1		╄	L		\dashv	4	_	
11																																			▙	<u> </u>		\dashv	_	\dashv	
12					-														-		-				-		-	-						-	╄	L		\dashv	4	\dashv	
13																					-						-						-		╄	▙		\vdash	\dashv	\dashv	_
14					-												-		-		-				-		-	-			-			-	╄	▙		\vdash	\dashv	\dashv	\dashv
15																	-		-												-		-		╀	\vdash		Н	\dashv	\dashv	_
16 17																	-		-		-		-				-				-			-	╆	⊨			\dashv	\dashv	\dashv
18																							-												+	-		\dashv	+	\dashv	\dashv
19																	-														-				+	-		\dashv	+	_	\neg
20																					T				1		T								+	-		H	\dashv	\dashv	\neg
21																																			+				7		_
22																			T	H				l									T	T	+	H		\sqcap	\dashv	\exists	
23																																			T			П	寸		
24																																						П	T		
25																																	L								
26																																									
27																																						oxdot			
28																																									
29																																			L	L		Ц			
30																																									

 Table 5: Pipeline time diagram, without data forwarding techniques.

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			 1			1	1												1					1																	
2 mil 5 22, \$ 23, \$ 4	_	INSTRUCTIONS					6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38 3	39 4	40
4 do Mi 3 5, \$ 5, \$ 1 5 to Mi 3 1, \$ 4, \$ 8 6 but \$ 5, \$ 5, top 7 5 w \$ 4, molf (\$0) 8 lw \$ 10, 0 (\$1) 10 11 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27	1																																								
4 do Mi 3 5, \$ 5, \$ 1 5 to Mi 3 1, \$ 4, \$ 8 6 but \$ 5, \$ 5, top 7 5 w \$ 4, molf (\$0) 8 lw \$ 10, 0 (\$1) 10 11 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27		dmol \$ 12, \$ 12, \$9	F	ŋ	R	R	M ₀	Ma	Ma	Mз	Mu	Ms	46	Μ	W																										
4 do Mi 3 5, \$ 5, \$ 1 5 to Mi 3 1, \$ 4, \$ 8 6 but \$ 5, \$ 5, top 7 5 w \$ 4, molf (\$0) 8 lw \$ 10, 0 (\$1) 10 11 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27	3	2020 5 9 4 9 612		F	F	٤	0	A	R	R	R	R	R	R	R	χ	M	W																							
9	4	dadi \$ 5, \$ 5, 1					F	F	F	F	F	4	F	F	F	V	χ	Μ	W																						
9	5	daddi 5 1, \$ 1,8														F	7	χ	Μ	W																					
10	6	bne \$ 6, \$ 5, 600															F	Ø	R	Χ	Μ	W																			
9	7	5 w \$ 9, mult (\$0)																F	F																						
9	8	(w \$12, 0 (\$1)																		F																					
11 12 13 14 15 16 17 18 19 <td< td=""><td>9</td><td>, , ,</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	9	, , ,																																							
12	10																																								
13 14 15 16 17 18 19 <td< td=""><td>11</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	11																																								
14 15 10 11 <td< td=""><td>12</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>	12																																								
15	13																																								
16 17 18 19 20 21 22 23 24 25 26 27 28 29	14																																								
17 18 19 20 21 22 23 24 25 26 27 28 29	15																																								
18 19 20 21 22 23 24 25 26 27 28 29	16																																								
19 20 21 22 23 24 25 26 27 28 29	17																																								
20 1	18																																								
21 22 23 24 25 26 27 28 29	19																																								
22 23 24 25 26 27 28 29	20																																								
23	21																																								
24 25 26 27 28 29	22																																					T	T		
25 3 3 4 5 6 6 6 6 7 6 7	23																																					T	T	T	
26 27 28 29	24																																					T	T		
27 28 29	25																																					T	T	T	٦
28 29 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3	26																																					T	T	T	٦
29	27																																					T	T	1	\exists
	28																																					T	\exists	1	
	29																																					T	\top	\exists	\exists
	30																																					7	十	寸	٦

Code for exercise 2.3:

```
.data
Α:
              .word 1, 3, 1, 6, 4
              .word 2, 4, 3, 9, 5
mult:
             0
       .word
       .code
       daddi
            $1, $0, A ; *A[0]
            $5, $0, 1 ; $5 = 1 ;; i
       daddi
       daddi $6, $0, 10 ; $6 = N ;; N = 10
       lw $9, 0(\$1) ; \$9 = A[0] ;; mult
       daddi $1, $1, 8 ;
loop:
      lw $12, 0($1); $12 = A[i]
      daddi $5, $5, 1 ; i++
       dmul
            $12, $12, $9 ; $12 = $12*$9 ;; $12 = A[i]*mult
            $1, $1, 8 ; moves to next A element
       daddi
            $9, $9, $12 ; $9 = $9 + $12 ;; mult = mult + A[i]*mult
       dadd
             $6, $5, loop ; Exit loop if i == N
       bne
             $9, mult($0); Store result
       SW
       halt
;; Expected result: mult = f6180 (hex), 1008000 (dec)
```

Code for exercise 2.4:

```
.data
                        1, 3, 1, 6, 4
Α:
                .word
                       2, 4, 3, 9, 5
                .word
mult:
        .word
        .code
                $1, $0, A
        daddi
                               ; $1 = &A[0]
                              ; $5 = i (comeca a 1 e vai até 4, para o loop correr 3x)
        daddi
                $5, $0, 1
        daddi
                $6, $0, 4
                $9, 0($1)
        1w
                $12, 8($1)
                               ; $12 = A[1]
        lw
                $13, 16($1)
                               ; $13 = A[2]
        1w
                $14, 24($1)
        1w
                               ; $1 = &A[1]()
        daddi $1, $1, 8
loop:
                $16, $12, $9 ; $16 = A[i]*mult
        dmul
        daddi
                $1, $1, 8
                               ; avança 1 elemento de A (fica em &A[i+1])
                $12, 16($1)
        1w
                $9, $9, $16
        dadd
                               ; mult = mult + A[i+1]*mult
                $17, $13, $9
                              ; $16 = A[i+1]*mult
        dmul
        daddi
                $1, $1, 16
                               ; Avança 2 elementos de A (fica em &A[i+3])
                $13, 8($1)
                               ; $12 = A[i+4]
        1w
        dadd
                $9, $9, $17
                               ; mult = mult + A[i+1]*mult
                $18, $14, $9
        dmul
                $5, $5, 1
        daddi
                $14, 16($1)
        lw
        dadd
                $9, $9, $18
                               ; mult = mult + A[i+2]*mult
                              ; Exit loop if index == N ;; 1^{\frac{1}{2}} e 2^{\frac{1}{2}} iter bne checks (2 stalls), 3^{\frac{1}{2}} iter bne falha -> guarda valor
        bne
                $6, $5, loop
        SW
                $9, mult($0)
                              ; Store result
        halt
```

Code for exercise 2.5:

```
.data
A:
              .word
                   1, 3, 1, 6, 4
                   2, 4, 3, 9, 5
              .word
mult:
       .word
              0
       .code
      daddi $1, $0, A ; *A[0]
      daddi
           $5, $0, 1 ; $5 = 1 ;; i
      daddi $6, $0, 10 ; $6 = N ;; N = 10
      lw $9, 0($1) ; $9 = A[0] ;; mult
      daddi $1, $1, 8 ;
loop:
             $12, 0($1); $12 = A[i]
      lw
      daddi $1, $1, 8
             $12, $12, $9 ; $12 = $12*$9 ;; $12 = A[i]*mult
      dmul
      daddi $5, $5, 1 ; i++
      bne
             $6, $5, loop ; Exit loop if i == N
      dadd
             $9, $9, $12 ; $9 = $9 + $12 ;; mult = mult + A[i]*mult
             $9, mult($0) ; Store result
       SW
      halt
  Expected result: mult = f6180 (hex), 1008000 (dec)
```