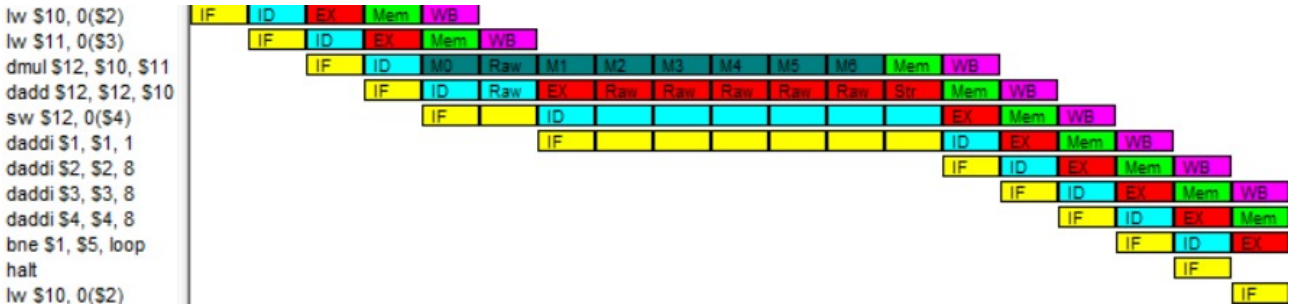
Third Lab Assignment: Instruction Level Parallelism

Number:	Name:
Maria Ramos	ist1105875
Guilherme Campos	ist1106906
Enzo Nunes	ist1106336

2.1 Simple execution, without data forwarding techniques

2.2 Application of data forwarding techniques



2.3 Source code optimization: minimization of data and structural hazards

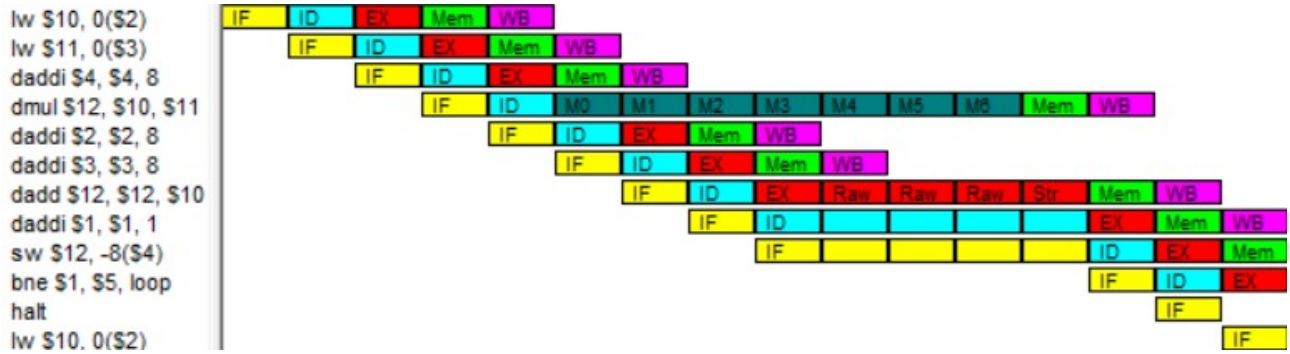
Copy of the new assembly program:

```
.data
A:      .word 1,2,3,4,5
        .word 6,7,8,9,10
        .word 11,12,13,14,15
        .word 16
B:      .word 11,22,33,44,55
        .word 66,77,88,99,100
        .word 111,122,133,144,155
        .word 166
C:      .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0

        .code
        daddi $1, $zero, 0 ; i = 0
        daddi $5, $zero, 16 ; value of N
        daddi $2, $zero, A
        daddi $3, $zero, B
        daddi $4, $zero, C

loop:   lw $10, 0($2)
        lw $11, 0($3)
        daddi $4, $4, 8
        dmul $12, $10, $11
        daddi $2, $2, 8
        daddi $3, $3, 8
        dadd $12, $12, $10
        daddi $1, $1, 1
        sw $12, -8($4)
        bne $1, $5, loop

        halt
```



2.4 Source code optimization: loop unrolling

Copy of the new assembly program:

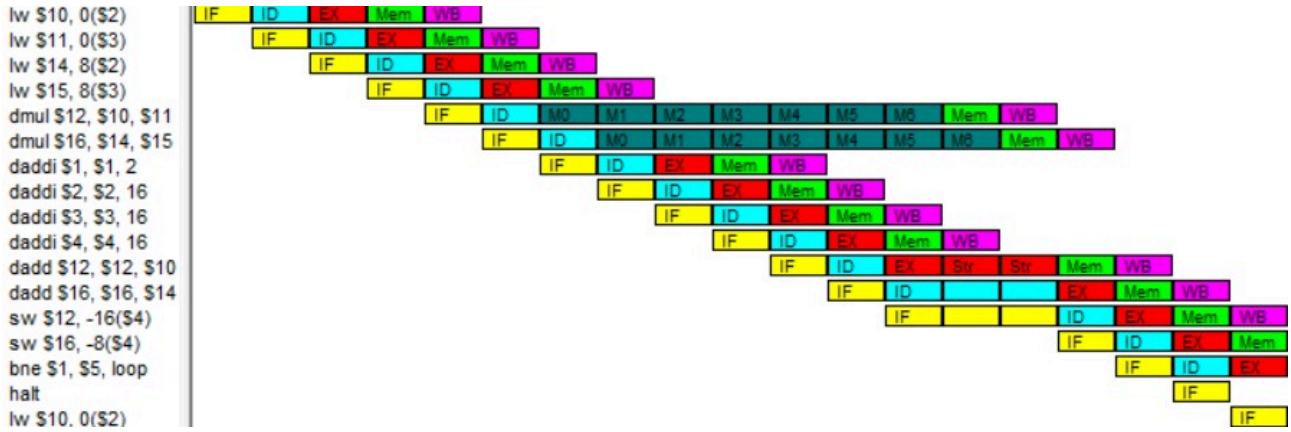
```
.data
A:      .word 1,2,3,4,5
        .word 6,7,8,9,10
        .word 11,12,13,14,15
        .word 16
B:      .word 11,22,33,44,55
        .word 66,77,88,99,100
        .word 111,122,133,144,155
        .word 166
C:      .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0

        .code
        daddi $1, $zero, 0 ; i = 0
        daddi $5, $zero, 16 ; value of N
        daddi $2, $zero, A
        daddi $3, $zero, B
        daddi $4, $zero, C

loop:   lw $10, 0($2)
        lw $11, 0($3)
        lw $14, 8($2)
        lw $15, 8($3)
        dmul $12, $10, $11
        dmul $16, $14, $15
         daddi $1, $1, 2
         daddi $2, $2, 16
        daddi $3, $3, 16
        daddi $4, $4, 16
        dadd $12, $12, $10

        dadd $16, $16, $14
        sw $12, -16($4)
        sw $16, -8($4)
        bne $1, $5, loop

        halt
```



2.5 Source code optimization: branch delay slot

Copy of the new assembly program:

```
.data
A:      .word 1,2,3,4,5
        .word 6,7,8,9,10
        .word 11,12,13,14,15
        .word 16
B:      .word 11,22,33,44,55
        .word 66,77,88,99,100
        .word 111,122,133,144,155
        .word 166
C:      .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0
        .word 0,0,0,0

        .code
        daddi $1, $zero, 0 ; i = 0
        daddi $5, $zero, 16 ; value of N
        daddi $2, $zero, A
        daddi $3, $zero, B
        daddi $4, $zero, C

loop:   lw $10, 0($2)
        lw $11, 0($3)
         daddi $4, $4, 8
        dmul $12, $10, $11
        daddi $2, $2, 8
        daddi $3, $3, 8
        daddi $1, $1, 1
        dadd $12, $12, $10
        bne $1, $5, loop
        sw $12, -8($4)

        halt
```