Pontifícia Universidade Católica do Rio Grande do Sul

TRABALHO FUNDAMENTOS DE SISTEMAS DIGITAIS

VETORES EM ASSEMBLY

II SEMESTRE

João Vitor Vogel
joao.vogel@edu.pucrs.br
Porto Alegre, Rio grande do Sul
Pontifícia Universidade Católica do RS - PUCRS

Abstract: Este trabalho tem como principal objetivo a criação de um código em assembly para a manipulação de dois vetores, consistindo basicamente na soma de ambos, após isso a subtração dos mesmos, sendo finalizado pela multiplicação do valor resultante da soma dos dois vetores oriundos da primeira parte pelo maior valor contido neles.

Foi necessário o envio do arquivo wave.do, pois foram utilizados mais waves do que os disponíveis no arquivo disponibilizado pelo professor.

Item 1 juntamente com Item 2

```
/*
 * Pseudo código referente ao tF de fundamentos de sistemas digitai
 * João Vitor Vogel 23/11/2023
 */
public class PseudoCodigo {
    // Simula os registradores
    static int t1 = 0; //contador para for
    static int t2 = 0; //variavel primeiro vetor
    static int t3 = 0; //variavel segundo vetor
    static int t4 = 0; //resultado de operacao
    static int t5 = 0;
    static int t6 = 0;
    static int t7 = 0
    static int t8 = 0;
    static int t9 = 0;

    //Declaração de variaveis na memoria
    static int[] A = {7, 15, 9, 22, 5, -3, -6, -17, -1, -21};
    static int[] B = {16, 13, 5, 9, 54, -4, -5, -8, -2, -12};
    static int[] C = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    static int[] D = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    static int mv = 0;  //maior valor dos vetores
    static int st = 0;  //soma total dos vetores
    static int sm = 0;  //resultado final da multuplicação

    public static void main(String[] args){
        for(t1=0; t1 < 10; t1++){
            soma();
        }
        for(t1=0; t1 < 10; t1++){
            subtrai();
        }
        t4 = 0;
        for(t1=0; t1 < 10; t1++){
            maiorValor();
        }
        mv = t4;
        t4 = 0;
        for(t1=0; t1 < 10; t1++){
            somaTodos();
        }
        st = t4;
```

```
        t2 = st; //resultado da soma
        t3 = mv; //maior valor, ou total de repetições
        t4 = 0;
        for(t1=0; t1 < t3; t1++){
            multiplica();
        }
        sm = t4;
    }
    public static void multiplica(){

        t4 = t4 + t2;
    }

    public static void somaTodos(){
        t2 = C[t1];
        t3 = D[t1];
        t4 = t4 + t2;
        t4 = t4 + t3;
    }
    public static void maiorValor(){
        t2 = C[t1];
        t3 = D[t1];
        if(t4 < t2){
            t4 = t2;
        }
        if(t4 < t3){
            t4 = t3;
        }
    }
    public static void soma(){
        t2 = A[t1];
        t3 = B[t1];

        t4 = t2 + t3;
        C[t1] = t4;
    }
    public static void subtrai(){
        t2 = A[t1];
        t3 = B[t1];

        t4 = t2 + t3;
        D[t1] = t4;
    } }
```

Item 3

```
# Variaveis utilizadas e seus valores

.data
a:   .word    7 15 9 22 5 -3 -6 -17 -1 -21     # Vetor A
b:   .word    16 13 5 9 54 -4 -5 -8 -2 -12     # Vetor B
c:   .word    0 0 0 0 0 0 0 0 0 0              # Vetor C
d:   .word    0 0 0 0 0 0 0 0 0 0              # Vetor D
mv:  .word    0                                # Maior valor
st:  .word    0                                # Resultado soma
sm:  .word    0                                # Resultado final
```

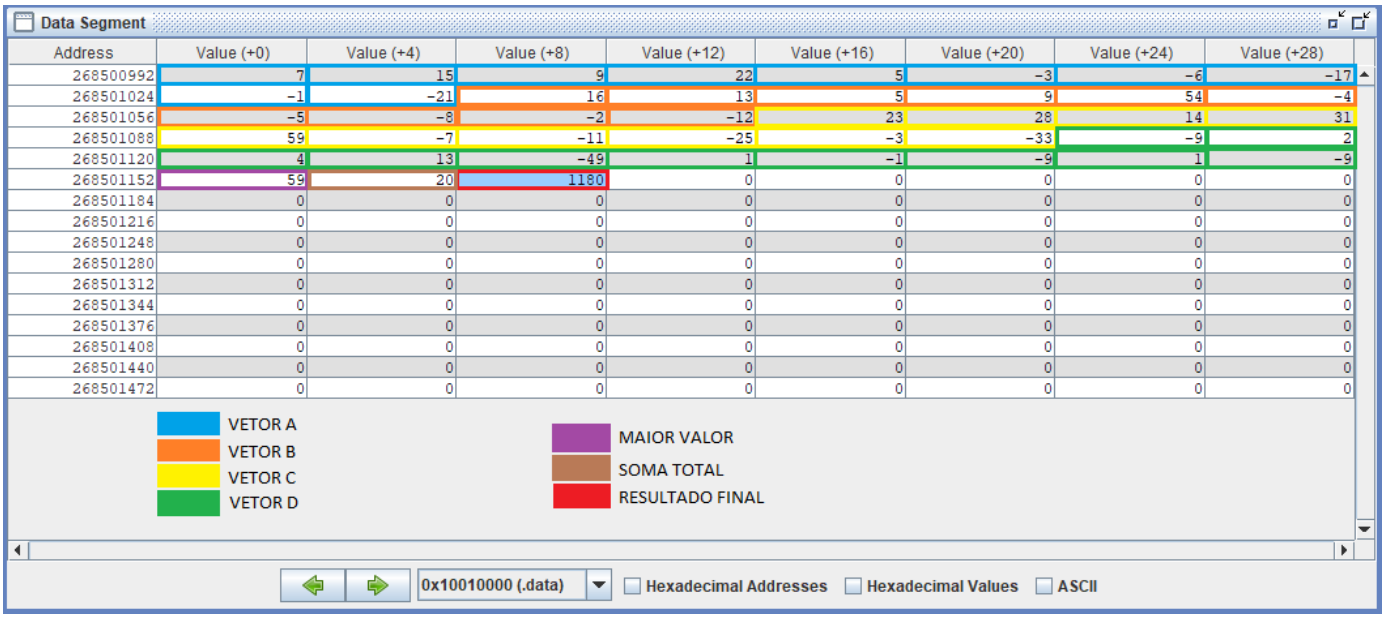
vetor C[i] = vetor A[i] + vetor B[i]
vetor D[i] = vetor A[i] - vetor B[i]

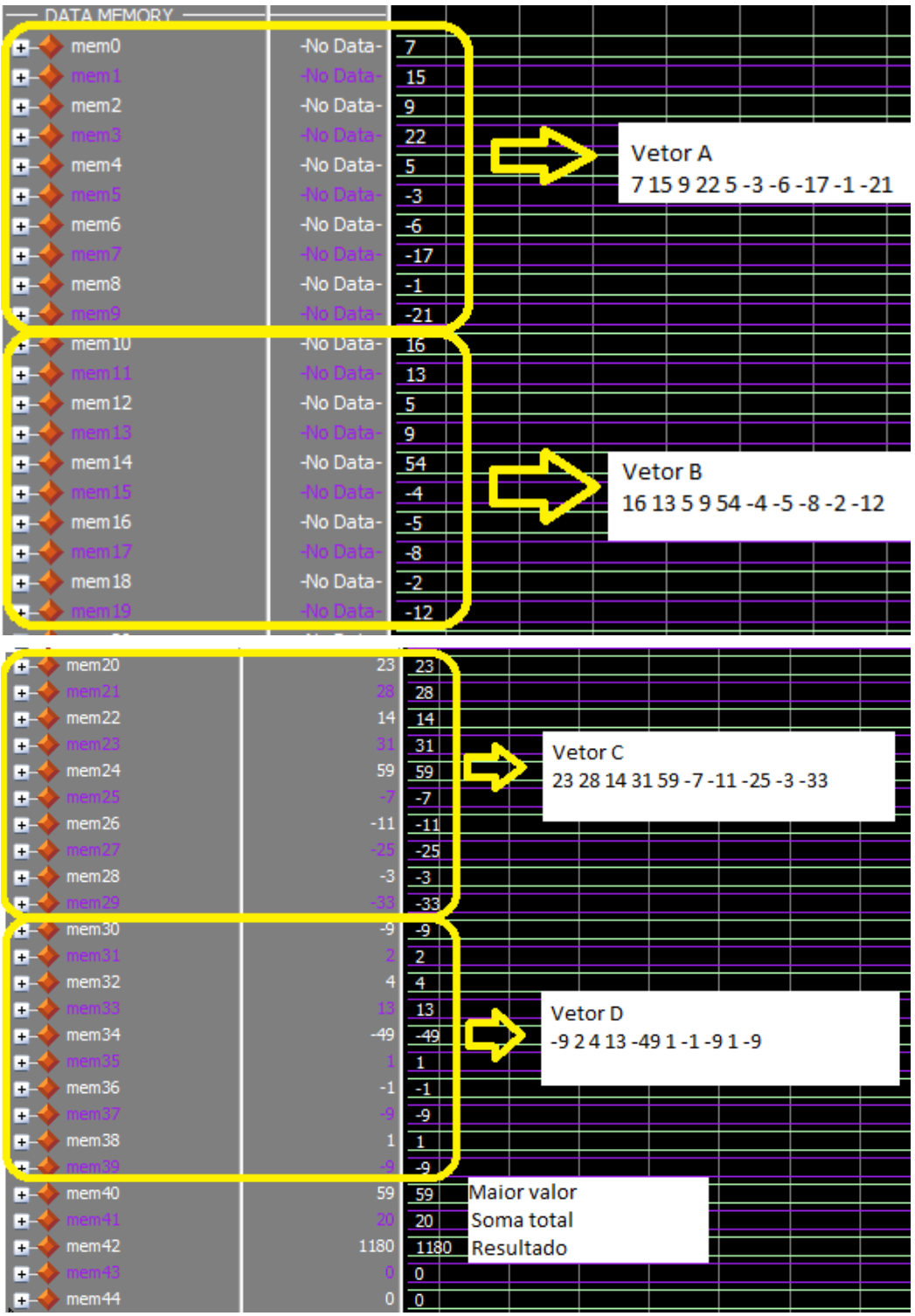
mv = maior valor encontrado em C ou D

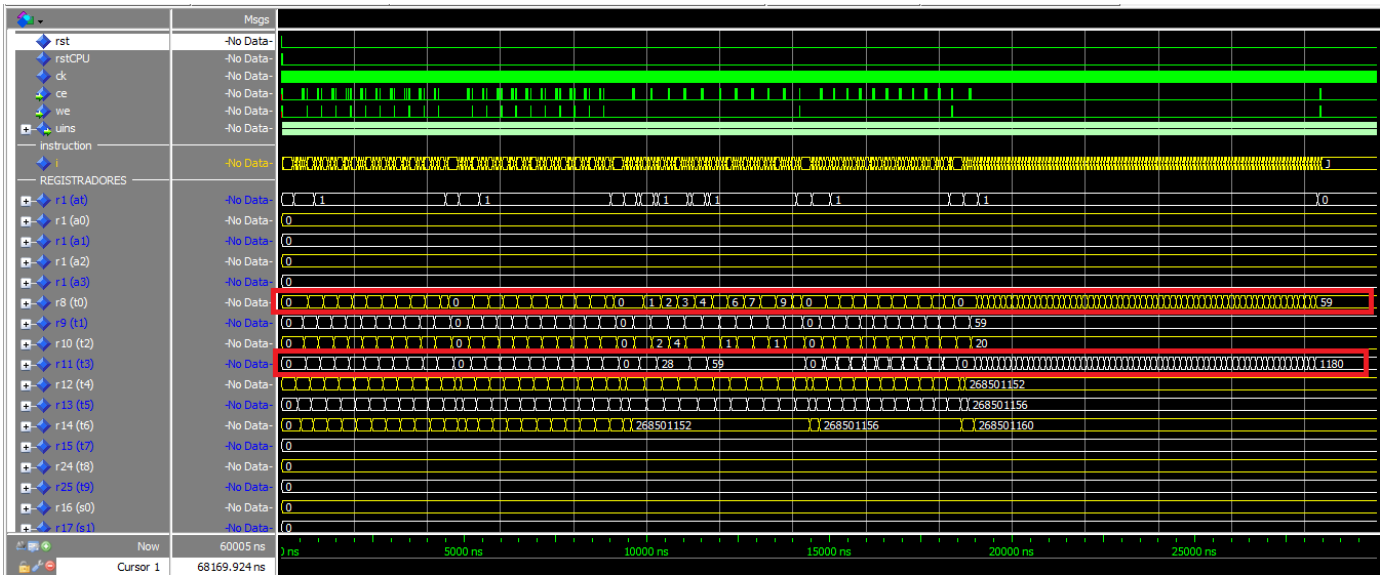
st = soma toal dos valores de C e D

sm = soma total * maior valor

Item 4




DATA MEMORY
mem0 -No Data- 7
mem1 -No Data- 15
mem2 -No Data- 9
mem3 -No Data- 22
mem4 -No Data- 5
mem5 -No Data- -3
mem6 -No Data- -6
mem7 -No Data- -17
mem8 -No Data- -1
mem9 -No Data- -21
Vetor A
7 15 9 22 5 -3 -6 -17 -1 -21
mem10 -No Data- 16
mem11 -No Data- 13
mem12 -No Data- 5
mem13 -No Data- 9
mem14 -No Data- 54
mem15 -No Data- -4
mem16 -No Data- -5
mem17 -No Data- -8
mem18 -No Data- -2
mem19 -No Data- -12
Vetor B
16 13 5 9 54 -4 -5 -8 -2 -12
mem20 23 23
mem21 28 28
mem22 14 14
mem23 31 31
mem24 59 59
mem25 -7 -7
mem26 -11 -11
mem27 -25 -25
mem28 -3 -3
mem29 -33 -33
Vetor C
23 28 14 31 59 -7 -11 -25 -3 -33
mem30 -9 -9
mem31 2 2
mem32 4 4
mem33 13 13
mem34 -49 -49
mem35 1 1
mem36 -1 -1
mem37 -9 -9
mem38 1 1
mem39 -9 -9
Vetor D
-9 2 4 13 -49 1 -1 -9 1 -9
mem40 59 59 Maior valor
mem41 20 20 Soma total
mem42 1180 1180 Resultado
mem43 0 0
mem44 0 0



devido ao tamanho da simulação fica difícil visualizar, mas basicamente é possivel perceber que o t0 está sempre realizando contagem de um e um, e que o t3 sempre armazena o resultado das operações