

FSD TP4 – ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

1 Orientações Gerais

O objetivo deste trabalho é exercitar e consolidar a compreensão dos alunos sobre programação em linguagem *assembly* focando na manipulação de vetores. Os alunos desenvolverão as habilidades de carregar, armazenar, manipular e processar vetores usando técnicas de programação em linguagem *assembly*.

Entrega do TP4	01/12/2023 – sexta-feira – 12h (meio-dia)
Este prazo de entrega é para permitir aos professores terem tempo de corrigir os trabalhos Não haverá extensão de prazo	

- a) O trabalho pode ser desenvolvido **individualmente** ou em **duplas**.

O número da especificação do trabalho a ser realizado é definido pelo resultado do resto da divisão por 10 da soma dos dígitos menos significativo da matrícula (sem dígito verificador) dos alunos que compõe a dupla:

0-1-2: especificação 1

3-4-5: especificação 2

6-7-8-9: especificação 3

Exemplo de dupla: {172221**5**-2 Alice e 2011558**7**-1 Bob}

Logo, o trabalho a ser realizado é o da **especificação 1, pois $7+5=12$ e $12\%10 = 2$.**

- b) Entregar um arquivo em formato **zip**, nomeando-o **nomeAluno1_nomeAluno2.zip**, com **3 arquivos: relatório.pdf, tp_GERAL.asm e tp_nro.asm**.
- c) Em **hipótese alguma** serão aceitos códigos fontes e/ou demais elementos **plagiados**. Em caso de ocorrência desta grave infração, a nota de todos os alunos envolvidos será sumariamente **zerada**.

2 Avaliação

Parte 1 [4 pontos]	Relatório (relatório.pdf), contendo 5 seções: <ul style="list-style-type: none"> (1) [1 ponto] pseudocódigo ou código (Python, C, Java...) do programa, explicando o mesmo; (2) [1 ponto] uma tabela que relacione as principais variáveis do pseudocódigo/código com os registradores da arquitetura MIPS; (3) [1 ponto] um exemplo da área de dados (diferente da apresentada), com a solução esperada; (4) [1 ponto] telas capturadas do simulador MARS, com explicação do que a tela apresenta;
Parte 2 [6 pontos]	Código em linguagem <i>assembly</i> , com comentários e corretamente indentado , usando apenas o conjunto de instruções visto em aula . Apresentar no relatório (seção 4) figuras da área de dados conforme exemplo apresentado posteriormente, correspondendo aos dados selecionados pelos alunos.

3 ESPECIFICAÇÕES DO TP4

ESPECIFICAÇÃO GERAL (para todos os alunos):

1. Crie dois vetores unidimensionais, A e B, cada um consistindo de n (onde $n \geq 8$) inteiros aleatórios com sinal **(devem ser utilizados pelo menos 2 valores positivos e 2 valores negativos)**.

2. Crie dois vetores C e D com o mesmo número de elementos (n) dos vetores A e B. Todas as posições dos vetores C e D devem ser inicializados com zero.
3. **Adição de Vetores** - $C(i) = A(i) + B(i)$
 - a. Crie uma sub-rotina em linguagem *assembly* que realize a adição de vetores.
 - b. A sub-rotina deve receber os vetores de entrada A e B e **armazenar** o resultado em um novo vetor C.
 - c. Use estruturas de looping apropriadas para iterar pelos vetores.
4. **Subtração de Vetores** - $D(i) = A(i) - B(i)$
 - a. Crie uma sub-rotina em linguagem *assembly* que realize a subtração de vetores.
 - b. A sub-rotina deve receber os vetores de entrada A e B e **armazenar** o resultado em um novo vetor D.
 - c. Use estruturas de looping apropriadas para iterar pelos vetores.
5. Siga para a **ESPECIFICAÇÃO** adequada ao **menor** dígito final da matrícula.

ESPECIFICAÇÃO 1 (para dígitos finais da matrícula **0-1-2**):

1. Crie uma sub-rotina que calcule o **produto escalar entre dois vetores** ($C \cdot D$)

$$PE = C \cdot D = \sum_{i=0}^n C_i D_i = C_0 D_0 + C_1 D_1 + C_2 D_2 + \dots + C_{n-1} D_{n-1}$$

2. Calcule o produto escalar dos vetores **C** e **D** que foram obtidos na ESPECIFICAÇÃO GERAL.
3. Salve o resultado em uma variável (**PE**) na memória.

Exemplo:

- Considerando: $C = \{10, 20, -5, 30\}$ $D = \{2, 3, 4, -8\}$
- $PE = (10 * 2) + (20 * 3) + ((-5) * 4) + (30 * (-8)) = 20 + 60 + (-20) + (-240) = -180$

ESPECIFICAÇÃO 2 (para dígitos finais da matrícula **3-4-5**):

1. Crie uma sub-rotina que realize a **multiplicação da soma dos valores positivos de dois vetores** ($MP \rightarrow$ multiplicação positivos)

$$MP = \left(\sum_{i=0}^n \max\{0, C_i\} \right) \cdot \left(\sum_{i=0}^n \max\{0, D_i\} \right)$$

2. Calcule a **multiplicação da soma dos valores positivos** dos vetores **C** e **D** que foram obtidos na ESPECIFICAÇÃO GERAL.
3. Salve o resultado em uma variável (**MP**) na memória.

Exemplo:

- Considerando: $C = \{10, 20, -5, 30\}$ $D = \{2, 3, 4, -8\}$
- $MP = (10 + 20 + 30) \times (2 + 3 + 4) = 60 \times 9 = 540$

ESPECIFICAÇÃO 3 (para dígitos finais da matrícula **6-7-8-9**):

1. Crie uma sub-rotina que calcule o **somatório de dois vetores multiplicado pelo valor máximo deles** ($SM \rightarrow$ máximo vezes somatório).

$$SM = \max(C, D) \cdot \left(\sum_{i=0}^n C_i + \sum_{i=0}^n D_i \right) = \max(C, D) \cdot (C_0 + C_1 + C_2 + \dots + C_{n-1} + D_0 + D_1 + D_2 + \dots + D_{n-1})$$

2. Calcule o **somatório dos vetores C e D**, que foram obtidos na ESPECIFICAÇÃO GERAL, **multiplicado pelo valor máximo deles**.
3. Salve o resultado em uma variável (**SM**) na memória.

Exemplo:

- Considerando: $C = \{10, 20, -5, 30\}$ $D = \{2, 3, 4, -8\}$
- $SM = 30 \cdot ((10 + 20 + (-5) + 30) + (2 + 3 + 4 + (-8))) = 30(56) = 1680$

4 Exemplo de execução para a especificação 1

Abaixo a estrutura da área de dados a ser utilizada, com um conjunto de valores em decimal utilizados para exemplificar a execução do problema (*especificação 1*) – **cada aluno/grupo terá necessariamente uma área de dados com valores diferentes do apresentado abaixo.**

.data

```
PE: .word 0
n: .word 7
A: .word 86 -324 56 -60 100 87 380
B: .word 25 -60 81 -89 100 456 124
C: .word 0 0 0 0 0 0 0
D: .word 0 0 0 0 0 0 0
```

Inicialmente, realizamos a soma e subtração de vetores para gerar os vetores C e D:

$C = \{111, -384, 137, -149, 200, 543, 504\}$

$D = \{61, -264, -25, 29, 0, -369, 256\}$

Após a execução da especificação geral, deve-se realizar a especificação do grupo (neste exemplo, a *especificação 1*):

$PE = 111*61 + (-384)*(-264) + 137*(-25) + (-149)*29 + 200*0 + 543*(-369) + 504*256 = 29058$

Seção 4 do relatório – telas capturadas do simulador MARS, com explicações

a) Ilustrar a área de dados antes de iniciar a execução

Data Segment										
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)		
0x10010000	0	7	86	-324	56	-60	100	87		
0x10010020	380	25	-60	81	-89	100	456	124		
0x10010040	0	0	0	0	0	0	0	0		
0x10010060	0	0	0	0	0	0	0	0		
0x10010080	0	0	0	0	0	0	0	0		
0x100100a0	0	0	0	0	0	0	0	0		

b) Ilustrar a área de dados ao final da execução

Data Segment										
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)		
0x10010000	29058	7	86	-324	56	-60	100	87		
0x10010020	380	25	-60	81	-89	100	456	124		
0x10010040	111	-384	137	-149	200	543	504	61		
0x10010060	-264	-25	29	0	-369	256	0	0		
0x10010080	0	0	0	0	0	0	0	0		
0x100100a0	0	0	0	0	0	0	0	0		

PE
n
A
B
C
D

Indicar nas 2 telas do MARS o valor do processamento (PE, MP ou SM), assim com os valores dos vetores A,B,C,D.

Dica de implementação de multiplicação:

```
int multiply(int a, int b){
    int acumula = 0;
    int vezes = b;

    // se B eh negativo, troca o sinal
    if(b < 0)
        vezes = 0 - vezes;

    // acumula A, B vezes
    while(vezes!=0){
        acumula = acumula + a;
        vezes = vezes - 1;
    }

    // se B eh negativo, trocamos o sinal do resultado
    if(b < 0)
        acumula = 0 - acumula;

    return acumula;
}
```