



UNIVERSIDADE FEDERAL DO CEARÁ
Campus de Quixadá
Prof. Thiago Werlley Bandeira da Silva
QXD0132- Arquitetura e Organização de Computadores I

AP2
2024.1

Nome: _____ Matrícula: _____

AP2 - ASSEMBLY

Tente implementar programas em Assembly para resolver os problemas abaixo, considerando as seguintes observações:

- Desenvolva os programas para funcionar na console do sistema operacional;
- Para realizar saída de dados, utilize uma forma simples de escrita na console, como a **função printf e scanf** no NASM;
- **LEMBRE-SE:** utilize o **Makefile** e coloque cada questão separado.
- Envie tudo zipado no SIGAA.
- Se atentem para o tempo de prova no SIGAA.

- [1 ponto] 1. Escreva um procedimento que recebe 2 parâmetros através da pilha de execução e retorna através do registrador EAX a soma dos valores. Todos registradores e flags devem ter seus valores preservados, com exceção de EAX que armazenará o resultado.
- [1 ponto] 2. Usando um **LAÇO e ENDEREÇO INDEXADO**, escreva o código que rotaciona os membros inteiros de 32-bit adiante em uma posição. O valor do final do array deve ir para a primeira posição. Por exemplo, o array [10, 20, 30, 40] será transformado em [40, 10, 20, 30].
- [1 ponto] 3. Escreva o seguinte código em assembly, modulado e com pilha. Mostre a impressão do resultado.

```
1 #include <stdint.h>
2 #include <stdlib.h>
3
4 int soma(int a, int size){
5     int sum;
6     for (sum = 0; size--> 0; )
7         sum += a++;
8     return sum;
9 }
10
11 int main(){
12     int s;
13     s = soma(1,5);
14     printf("soma eh:%d \n", s);
15     return 0;
16 }
```

Observação: O loop continua enquanto o valor de *size* for diferente de 0.
Quando *size* chega a 0, a expressão "*size--> 0*" se torna false, e o loop para.
Em outras palavras, o loop itera *size* vezes e termina quando *size* chega a zero.

- [1 ponto] 4. Faça um procedimento que receba um número inteiro e positivo *N* como parâmetro e retorne a soma dos números inteiros existentes entre 1 e *N* (inclusive).
Seu procedimento deve respeitar a seguinte documentação:

Nota: _____

```

1 ;-----
2 PROC:
3 ;
4 ; Calcula e retorna a soma dos inteiros de 32-bit de 1 ate N.
5 ; Receives: EAX o valor de N. Sem sinal.
6 ; Returns: EAX = soma, e as flags de status(Carry,
7 ; Overflow, etc.) sao modificadas.
8 ; Requires: nada
9 ;-----

```

- [1 ponto] 5. Faça um programa que receba o preço de um produto e mostre, de acordo com a tabela a seguir, a sua classificação, usando escolha dirigida por tabela.

Preço	Classificação
Até R\$80,00	Barato
Entre R\$80,00 e R\$120,00(inclusive)	Normal
Entre R\$120,00 e R\$200,00(inclusive)	Caro
Maior que R\$200,00	Muito Caro

- [1 ponto] 6. Faça um programa que receba uma matriz 6x3, identifique e mostre:

- o maior elemento da matriz e sua respectiva posição, ou seja, linha e coluna;
- o menor elemento da matriz e sua respectiva posição (linha, coluna)

Dica, implemente um procedimento que recebe linha x coluna e retorna o *offset* do elemento na memória.

- [2 pontos] 7. Crie um programa que receba 3 valores (obrigatoriamente maiores que zero), representando as medidas de três lados de um triângulo. Elabore procedimentos para:

- determinar se esses lados formam um triângulo (sabe-se que, para ser triângulo, a medida de um lado qualquer deve ser inferior ou igual à soma das medidas dos outros dois)
- determinar e mostrar o tipo de triângulo (equilátero, isósceles ou escaleno), caso as medidas formem um triângulo.

- [2 pontos] 8. Faça um programa que receba o ano de nascimento de uma pessoa e o ano atual, calcule de acordo com as restrições e mostre:

- a idade da pessoa em anos
- a idade da pessoa em meses
- a idade da pessoa em dias

Restrições:

- As únicas operações aritméticas permitidas são soma e subtração
- Qualquer tipo de salto pode ser utilizada
- Você deve escolher o tamanho das variáveis para que seja possível resolver o problema.