

# UNIVERSIDADE ESTADUAL DE MARINGÁ

## DEPARTAMENTO DE INFORMÁTICA

### Programação para o Interfaceamento de Hardware e Software

Prof. Ronaldo – Ano Letivo 2021

Prova – 14/12/2021

Desenvolver um programa montável, linkável e executável em Gnu Assembly 32 bits para ler 2 vetores de números inteiros, V1 e V2, de tamanhos tam1 e tam2, e concatená-los em um único vetor V3, de tamanho tam3, mostrando-o na tela. Depois, o programa deverá calcular a média real (em double) dos elementos do vetor, mostrando na tela o resultado. As seguintes condições devem ser satisfeitas: **1)** A leitura dos vetores deverá ser feita por uma mesma função chamada leiaVetor, a qual deverá ser chamada com call/ret, sendo que o nome do vetor deverá ser passado como parâmetro via pilha. Em seu funcionamento, a função leiaVetor deverá informar qual vetor está sendo lido; A função leiaVetor deverá solicitar do usuário o número de elementos a ser lido e com isso alocar dinamicamente o espaço do vetor. A função deverá retornar em %eax o número de elementos lidos e o endereço do vetor deverá ser retornado no topo da pilha; A função leiaVetor deverá garantir que os elementos dos vetores sejam lidos em **ordem crescente**, mas veja, não é para ler todo o vetor e depois ordená-lo, mas para já ler cada elemento na ordem certa e se o usuário digitar na ordem errada, o elemento deverá ser “resolicitado”; A função leiaVetor deverá informar qual o elemento que está sendo lido. Elementos repetidos poderão ser aceitos durante a leitura; **2)** Após a leitura, os vetores V1 e V2 deverão ser mostrados na tela por uma mesma função chamada mostraVetor, chamada com call/ret; A função mostraVetor deverá informar qual vetor está sendo mostrado e os parâmetros necessários para isso devem ser passados todos pela pilha; **3)**

Após os 2 vetores serem mostrados, estes deverão ser percorridos e concatenados (juntados) em um mesmo vetor V3, por meio da função `concatenaVetores`, de tal forma que o vetor V3 esteja em **ordem decrescente** e os elementos repetidos tenham sido removidos, ficando apenas a primeira ocorrência de cada elemento; As informações dos vetores devem ser passadas para o `concatenaVetores` via pilha. O tamanho do vetor V3 deve ser retornado em `%eax`. Observem que não é para ordenar o vetor V3 depois de montado, mas durante a concatenação, ou seja, o v3 deve ser montado já com os elementos um a um em ordem decrescente. Mostre como ficou o vetor V3 usando a função `mostraVetor`;

**4)** A média aritmética dos elementos do vetor V3 deve ser calculada pela função `calculaMedia` e o valor dela deve ser retornado como `double` no topo da pilha. O endereço e o tamanho do V3 devem ser passados pela pilha. A média resultante deve ser mostrada na tela. Implemente a possibilidade de reexecução.

**OBS:** entregue o código fonte no Moodle. Observe o horário limite de término. Não será aceito envio por email e nem após o término do prazo. Códigos parecidos entre alunos serão desconsiderados. Os códigos devem permitir montagem, linkagem e execução, ou seja, os códigos devem “rodar”. Não façam uso de `makefile`. Não compactem o código fonte, entregue-o ao natural na linguagem Assembly. Instruções diferentes das passadas em aula serão aceitas de forma restrita mediante explicação do seu funcionamento. Documente o código explicando como as funções foram planejadas em termos de funcionamento e de passagem de parâmetros. Faça tela de abertura e imprima mensagens de apresentação no início de cada função. Tabule o código.

**Valoração:** leitura (1,5), mostragem (1,5), concatenação (3,0), calculo da média (3,0) e restante com reexecução (1,0).