

## Linguagens de Montagem

Prof. Daniel Pedronette

### Lista de Exercícios 2

#### Instruções Lógico-Aritméticas

1. Escreva um programa em Assembly que declare uma estrutura de memória de 4 posições inicializando-as com o valor 1. Em seguida, realize as seguintes transferências de memória:

- Valor 2 para a segunda posição da estrutura;
- Valor 4 para a terceira posição da estrutura;
- Valor 8 para a última posição da estrutura.

Retorne como saída do programa a soma de todas as posições do vetor.

Utilize as instruções INC e ADD.

2. Transcreva as expressões numéricas abaixo para um programa em Assembly:

a.)  $X = A + B * C - D * F + G - H$

b.)  $X = H - B * (-D) + (-C) * G + A$

Retorne o valor de X para o sistema operacional. Atribua às incógnitas A..H a classificação da respectiva letra no alfabeto. Por exemplo, o resultado da primeira expressão deve ser -18 e deve retornar 238 (Por quê?).

3. Considere um byte de memória identificado pelo label “var” e definido pela diretiva DB.

a.) Escreva um programa em Assembly que calcule e retorne o complemento de 2 do valor contido em “var”.

b.) Escreva um programa em Assembly que calcule e retorne o valor de  $var * (-1)$ , utilizando instruções de multiplicação.

4. Escreva um programa em Assembly que calcule o valor  $X = 5! - (5 + 4 + 3 + 2 + 1)$

5. Considere dois vetores de números inteiros  $P = [2, 3, 5]$  e  $N = [8, 6, 7]$ , em que N representa notas de 3 avaliações e P seus respectivos pesos. Considerando média 5 como nota final mínima para aprovação, escreva um programa em Assembly inicialize tais vetores e:

a.) Retorne 0, caso o aluno tenha sido reprovado e qualquer outro valor caso tenha sido aprovado.

b.) Exiba “Aprovado: 1” caso o aluno tenha sido aprovado ou “Aprovado: 0” caso o aluno tenha sido reprovado.

Obs: Utilize apenas as instruções lógico-aritméticas e de transferência de memória já discutidas. Teste outros valores para inicialização dos vetores de notas e de pesos.