

Trabalho 2

Entregar o trabalho pelo Aprender. Trabalho pode ser feito em duplo ou individualmente. Deve ser entregue o código e um arquivo readme com detalhes de funcionamento e compilação do programa.

Especificação - Carregador

A. Detalhes de Implementação

Realizar um programa chamado “carregador”. Este programa deve ser feito da seguinte maneira:

- 1 – programa principal deve ser feito em C e deve chamar uma ou mais funções em Assembly NASM IA-32
- 2 – O programa em C deve interpretar a linha de comando e passar parâmetros pela pilha para uma função em Assembly que deve fazer os cálculos requeridos
- 3 – Outra função em assembly que também recebe qualquer parâmetro pela pilha deve imprimir todas as mensagens nas telas. Os strings podem ser variáveis locais ou globais.
- 4 – NENHUM parâmetro lido pelo programa em C da linha de comando pode ser salvo como variável GLOBAL. Deve ser sempre passado as funções como parâmetros pela pilha.

Não seguir estas indicações implica na nota ZERO do trabalho.

B. Funcionamento do programa é da seguinte forma.

O Usuario chama o programa pela linha de comando usando o seguinte formato:

```
./carregador 125 100 500 4000 300 20000 125 30000 345
```

Após o nome do programa o primeiro número indica é o tamanho de um programa fictício a ser carregado na memória. Depois pode ter até 4 pares de números separados por espaço. O primeiro número do par é o endereço de memória que inicia um bloco livre na memória e o segundo e a quantidades de unidades livres nesse bloco.

No exemplo anterior. O programa teria 125 unidades de tamanho. E temos 4 blocos onde o primeiro começa no endereço 100 e tem 500 unidades disponiveis. O segundo bloco começa no endereço 4000 tendo 300 unidades livres. E o terceiro bloco começaria no endereço 20000 e assim.

O carregador deve verificar se o programa fictício cabe em algum bloco por inteiro. Caso não couber deve tentar dividir o programa em pedaços sempre utilizando o máximo do tamanho possível dos blocos na ordem indicada. Caso mesmo assim não couber deve dar mensagem que não tem espaço em memória para carregar esse programa. Caso o programa ficticio couber em um bloco, ou dividido em blocos o carregador deve indicar a lista de blocos, o endereço inicial e final de cada parte do programa em cada bloco. Para isso, utilize mensagens de texto para poder interpretar facilmente sua saída.

Nao é permitido utilizar a IO.MAC