

Instrumento Complementar de Avaliação 02

30 pontos

Leia atentamente TODO o documento

Desenvolva um programa que ordene uma lista simplesmente encadeada.

Ao iniciar, seu programa vai lendo todos os números digitados pelo usuário e inserindo no final de uma lista simplesmente encadeada, que está inicialmente vazia. A leitura dos números termina quando o usuário digita um número negativo (Figura 1).

```
4
7
1
800
4
9
-67
```

Figura 1

Assim que um número negativo for digitado, o programa deve exibir todos os números da lista seguindo o formato da Figura 2.

```
4->7->1->800->4->9->
```

Figura 2

Em seguida o seu programa deve colocar em ordem crescente todos os números da lista encadeada, e exibir novamente a lista no final (Figura 3).

```
1->4->4->7->9->800->
```

Figura 3

REGRAS:

1. A lista deve ser simplesmente encadeada.
2. A lista pode conter números repetidos.
3. Assuma que a lista terá no mínimo 1 número e no máximo 1000.
4. Não se pode usar vetor e nem qualquer outra estrutura auxiliar (lista, pilha, fila, etc.) para ajudar na ordenação dos números.
5. Para ordenar não é permitido trocar os números. Deve-se trocar os nós de posição.
6. Uma vez inserido todos os números na lista, não é permitido apagar (free) e nem recriar nós (malloc).

7. Seu programa deve ser capaz de ler os números de um arquivo usando o redirecionamento e entrada do Linux.
Ex: ./ordena < dados.txt

Bônus Surpreenda:

Este ICA terá até 5 pontos de bonificação. Para isso o grupo precisa desenvolver algo a mais, algo que não foi previsto na especificação e que me surpreenda. É importante que seja algo diferente, inovador, único, que difere seu trabalho dos demais grupos.

Componentes:

A resolução deste trabalho deverá ser feita **obrigatoriamente** em grupos de **no mínimo 02 (duas) e no máximo 03 (três)** pessoas.

Data da Entrega:

O trabalho deve ser entregue até as **23:59** horas do dia **03/12/2018 (segunda-feira)**.

Forma de Entrega:

- Seu programa deve ser composto de 03 arquivos: um arquivo .h chamado **lista.h**, onde será descrito a estrutura de dados da sua lista e as assinaturas de todas as funções necessárias para manipular essa lista; um arquivo .c chamado **lista.c**, onde será implementado todas as funções da lista que foram definidas no .h; e um outro arquivo .c chamado **ordenacao.c**, onde fica a implementação da lógica do programa que ordena a lista de números.
- Compacte o código fonte do programa. O nome do arquivo compactado deverá ser **ica2.zip**.
- Envie o arquivo compactado, via Moodle, na atividade criada para este ICA.

Observações Importantes:

- Os trabalhos serão verificados automaticamente por uma ferramenta de detecção de plágio. Em caso de detecção de cópia (parcial ou integral), todos os envolvidos recebem nota **ZERO**. Em outras palavras, tanto os alunos que copiaram quanto o que deixou copiar recebem **ZERO**.
- Envie o trabalho no prazo especificado e no formato especificado. Trabalhos recebidos fora do prazo ou em formato inadequado serão **penalizados**.
- Os trabalhos serão **compilados** e **testados** no Sistema Operacional **Linux**.
- Trabalhos que **NÃO** compilam **NÃO** serão avaliados e receberão nota **ZERO**.
- Os programas serão avaliados pela sua conformidade com a especificação, correção durante a execução e pelo estilo de programação. Serão observados particularmente se os programas possuem os comentários apropriados, se usam nomes significativos para as variáveis e procedimentos, e se o código está endentado corretamente.

Distribuição de pontos:

O valor total deste trabalho é **30 pontos**, divididos da seguinte forma:

- *Código-fonte: 25 pontos*
Corretude, bugs, implementação do programa.
- *Organização, endentação e comentários: 5 pontos*
Organização e endentação do código bem como comentários adequados

Erratas ou Alterações:

Qualquer correção ou alteração da especificação do trabalho será feita em Sala de Aula e via e-mail.

Dúvidas?

Via e-mail (amonteiro@ucv.edu.br) ou durante as aulas.