

SSC0603 – Estrutura de Dados I – 2025-2 (ED1)

Professor responsável: Fernando Santos Osório

Semestre: 2025/2

Horário: Quinta 13h/14h (Prova P1)

Wiki: SSC-603-2024(FOsorio)

Web: <http://www.icmc.usp.br/~fosorio/>

DATA PROVA: 09/10/2025 - PROVA P1

NRO. USP: <Colocar o seu NUSP no programa fonte> COMO COMENTÁRIO no CÓDIGO

NOME : <Colocar o seu Nome no programa fonte> COMO COMENTÁRIO no CÓDIGO

>> COLOCAR SEU NOME E NRO. USP COMO COMENTÁRIO DO PROGRAMA FONTE ENTREGUE!

PROVA PROVA P1 – SSC0603 – ED1

Q1 (Questão única). Implemente o seguinte programa descrito abaixo de acordo com o especificado.

O programa deve ser enviado para o RunCodes da Disciplina SSC0603 (Prova P1)

RunCodes SSC0603 - <https://runcodes.icmc.usp.br/offerings/view/123> - Regras da Prova no final deste texto.

O programa da prova é um “estudo climático” (ciência de dados) sobre as temperaturas máximas de uma cidade em cada dia do ano, fornecidas em um arquivo texto (“dados.txt”). Analisando este arquivo, vamos determinar a temperatura máxima (TMax) e quantos dias tiveram temperaturas próximas a esta temperatura máxima (variação Delta de “n” graus). O programa deve USAR LISTAS ENCADEADAS DINÂMICAS (TAD visto em aula ou outro qualquer desde que tenha alocação dinâmica de nodos e a lista seja encadeada com ponteiros).

Descrição: Criar um programa “modular” (com sub-rotinas e TAD LDED [preferencial] ou LDES, conforme descrição mais abaixo), que realize as seguintes tarefas:

(1) Ler um arquivo texto de entrada “dados.txt” que possui os seguintes dados:

Dia Mês Ano TempMax

=> Dia, mês e ano são inteiros separados por espaço

=> TempMax é a temperatura máxima deste dia (pto. flutuante, 2 casas dec.)

Os dados são 4 números por linha do arquivo separados por espaço.

A última linha de dados possui o valor -1 para o dia, mês, ano e temp.

Exemplo: Entrada Arquivo “dados.txt”

```
01 01 2024 1.00
02 01 2024 2.00
03 01 2024 3.00
04 01 2024 4.00
05 01 2024 5.00
06 01 2024 6.00
07 01 2024 7.00
08 01 2024 8.00
09 01 2024 9.00
10 01 2024 10.00
-1 -1 -1 -1.00
```

(2) Você deve ler estes dados e inserir em uma lista encadeada na ordem que são fornecidos, no exemplo acima, do dia 1-1-2-24 ao dia 10-1-2024. A linha final com -1 não deve ser armazenada.

(3) Você deve exibir os seguintes dados na tela: um dado em cada linha como descrito abaixo

3.1 Total de dados lidos do Arquivo: **10**

3.2 Temperatura máxima da lista de dados: **10 1 2024 10.00** (exibir exatamente assim, 2 casas TMax)

3.3 Considerando uma variação (Delta) de 4 graus, valor fixo e sempre o mesmo de 4.00 em relação a temperatura máxima, quantos dias nesta lista foram acima de TMax-Delta ? No exemplo: **5** (Dias com 6.00 graus ou mais de temperatura, incluindo o dia de TMax – Considera toda lista)

3.4 Considerando uma variação (Delta) de 4 graus, valor fixo e sempre o mesmo de 4.00 em relação a temperatura máxima, quantos dias seguidos ANTES de TMax foram acima de TMax-Delta ?

No exemplo acima foram: **4** (de 06 01 2024 a 09 01 2024)

3.5 Considerando uma variação (Delta) de 4 graus, valor fixo e sempre o mesmo de 4.00 em relação a temperatura máxima, quantos dias seguidos DEPOIS de TMax foram acima de TMax-Delta ?

No exemplo acima foram: **0** (pois não temos dias depois do dia do TMax)

=> Procure implementar todas estas 5 saídas (5 linhas de saída na tela), mas se não conseguir, implemente o que for possível!

(4) O programa termina depois de mostrar os resultados acima:

```
10  
10 1 2024 10.00  
5  
4  
0
```

Exemplo mais completo: 1 mês de dados (entrada) – Caso de Teste 1

```
01 01 2024 1.00  
02 01 2024 2.00  
03 01 2024 3.00  
04 01 2024 4.00  
05 01 2024 5.00  
06 01 2024 6.00  
07 01 2024 7.00  
08 01 2024 8.00  
09 01 2024 9.00  
10 01 2024 10.00  
11 01 2024 9.00  
12 01 2024 8.00  
13 01 2024 7.00  
14 01 2024 6.00  
15 01 2024 5.00  
16 01 2024 5.00  
17 01 2024 5.00  
18 01 2024 5.00  
19 01 2024 4.00  
20 01 2024 3.00  
-1 -1 -1 -1.00
```

Saida do programa na tela: Caso de Teste 1

```
20  
10 1 2024 10.00  
9  
4  
4
```

BOA PROVA!!!

REGRAS EM RELAÇÃO REALIZAÇÃO DESTA PROVA

1. A PROVA É INDIVIDUAL com consulta Papel (Livros, papel impresso ou escrito) e formato Digital (Internet, Wiki, Pendrive), porém SEM CONSULTAR ou SE COMUNICAR COM HUMANOS ou QUALQUER OUTRA FORMA DE VIDA TERRESTRE ou EXTRA-TERRESTRE!
Não podem usar de formas de comunicação com pessoas externas, além do professor, referente a prova, seja por celular (manter desligado/guardado), por e-mail, por whatsapp, por mensagens ou fóruns (“ao vivo”), com colegas, etc.
NÃO É PERMITIDO O EMPRÉSTIMO DE MATERIAL (Cadernos, Anotações, Livros, PenDrive, etc).
 2. A PROVA TEM QUE SER FEITA NO LABORATÓRIO 8.113 / 8.115 USANDO OS COMPUTADORES PCs DO LABORATÓRIO (SEM VPN, SEM ACESSO REMOTO!)
O ENDEREÇO DE IP USADO (Endereço de rede) PARA O ENVIO DA PROVA TEM QUE SER DAS MÁQUINAS DO LABORATÓRIO 8.113/8.115. AS MÁQUINAS SÃO MONITORADAS
>> As máquinas do Lab. 8.113 e 8.115 estão sendo monitoradas << Sniff, sniffer, cuidado WireShark 😊
 3. A PROVA DEVE SER ENTREGUE via RunCodes.icmc (pode submeter várias vezes)
ENVIAR O ARQUIVO ZIP do MAKEFILE – O envio de “.c” é “aceito”, mas tem um desconto na nota.
Entrega é via <https://runcodes.icmc.usp.br/> Disciplina SSC0603 – Prova P1
 4. EM CASO DE PROBLEMAS no RunCodes.icmc pode ser entregue por E-Mail para:
fosorio @ icmc.usp.br com cópia (Cc:) para fosorio @ gmail.com
Com assunto/subject: **Prova-P1 <Seu_Nome> <Nro_USP>**
ANEXANDO O PROGRAMA FONTE (Zip do Makefile ou se não conseguir, o “.C” e o Project “.CBP”)
NÃO ANEXAR EXECUTÁVEIS! JAMAIS ENVIE EXECUTÁVEIS POR E-MAIL
NÃO ANEXAR OBJ, BIN e EXE!!! Seu e-mail não será recebido se for anexado bin, obj ou exe!
 5. A PROVA USUALMENTE DEVE SER REALIZADA USANDO O CODEBLOCKS DO LABORATÓRIO, e depois enviada (testada com os casos de teste) do RunCodes (USUAL).
 6. AO ENTREGAR A PROVA NO RUNCODES ou POR E-MAIL O(A) ALUNO(A) CONCORDA COM ESTAS REGRAS E SE COMPROMETE A FAZER A PROVA INDIVIDUALMENTE!
-

FIM