

10 Programação em Linguagem C

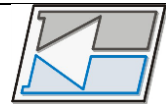
SEGUNDA AVALIAÇÃO – 2022/1

OBSERVAÇÕES IMPORTANTES

1. ESTA AVALIAÇÃO CONTÉM 4 QUESTÕES E 4 PÁGINAS.
2. **Antes de iniciar a avaliação coloque seu material de consulta no computador e depois desconecte o cabo de rede.**
3. Todos os códigos **deverão** apresentar um cabeçalho contendo a informação sobre o autor do programa e o que o programa faz. Por exemplo:

```
/*  
* Autor: Fulano de Tal  
* Este programa responde se um ano fornecido pelo usuário é um ano  
* bissexto  
*/
```
4. Programas que não compilam/interpretam não serão considerados. Portanto entregue apenas os itens que estiverem funcionando e respeitando o solicitado.
5. Além dos resultados pedidos em consonância com **o solicitado** serão avaliados:
 - Uso apropriado do tipo de variável para cada situação
 - Endentação
 - Tamanho do código
6. Não podem ser utilizadas variáveis globais, ou seja, variáveis declaradas fora de todas as funções do programa

ENTREGA NO SIGAA IMPRETERÍVELMENTE ATÉ ÀS 11h 40 min.



QUESTÃO 1 – Seja a seguinte struct C:

```
struct tempo{  
    int hora;  
    int min;  
    int seg;  
}
```

Faça um programa para calcular o tempo transcorrido entre duas marcações subsequentes de um cronômetro, de acordo com as seguintes especificações:

- Declare no programa 3 variáveis do tipo struct tempo para armazenar o tempo inicial, o tempo final e o tempo transcorrido:

```
struct tempo ti, tf e delta_t;
```

- Os instantes de tempo inicial (t_i) e final (t_f) da marcação do cronômetro devem entrar pelo teclado no programa principal no formato <hora> <min> <seg>;

- O programa principal passa **por referência** para uma função as variáveis t_i , t_f e δt .

- A função calcula o tempo transcorrido δt entre os instantes t_i e t_f ;

- O programa principal apresenta o tempo transcorrido no formato <hora>:<min>:<seg>

- O protótipo da função deve se;

```
void Elapsed_time( struct tempo*, struct tempo*, struct tempo*);
```

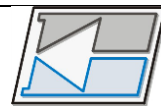
Exemplo de funcionamento:

Entre com a marcação inicial do cronometro em horas minutos e segundos <h> <min> <seg>: **2 25 45**

Entre com a marcação final do cronometro em horas minutos e segundos <h> <min> <seg>: **4 15 20**

O intervalo de tempo transcorrido foi de:

1 h 49 min 35 seg



QUESTÃO 2 – Elabore um programa para aumentar uma matriz $m \times n$ (máximo 10×10) acrescentando-lhe uma coluna, de acordo com as seguintes especificações:

- A matriz e um vetor devem ser declarados no programa principal;
- O programa principal pergunta o número de linhas **m** e colunas **n** da matriz;
- Uma função recebe a matriz por referência e as suas dimensões e gera seus valores seguindo a seguinte lei de formação:

$$A[i][j] = (5-i) * (j+1)$$

- Outra função (`Imprime_matriz`) recebe a matriz gerada por referência e as suas dimensões e a imprime na tela;
- O usuário informa pelo teclado os valores da coluna a ser inserida na matriz, valores esses que são armazenados no vetor declarado;
- O programa principal envia para uma função as dimensões da matriz, a matriz e o vetor, ambos por referência, função esta que vai acrescentar o vetor como uma nova coluna da matriz.
- A função `Imprime_matriz` imprime a matriz atualizada.

Exemplo de funcionamento (comandos do usuário em negrito):

Entre com o número de linhas e colunas da matriz
<linhas> <colunas>: **3 3**

Matriz gerada

5 10 15

4 8 12

3 6 9

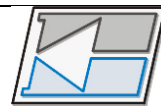
Entre com a nova coluna da matriz: **10 5 8**

Matriz aumentada:

5 10 15 **10**

4 8 12 **5**

3 6 9 **8**



QUESTÃO 3 – Dado um número **inteiro positivo e par** n lido do teclado, elaborar um programa para calcular e imprimir o valor das seguintes somas :

(a) **Fórmula de Leibniz para $\pi/4$**

$$soma = \sum_{i=0}^n \frac{(-1)^{ni}}{2i+1}$$

(b)

$$soma = -\log_{10}\left(1 + \frac{1}{2}\right) + \log_{10}\left(1 + \frac{3}{2}\right) - \log_{10}\left(1 + \frac{5}{2}\right) + \dots - \log_{10}\left(1 + \frac{2n-1}{2}\right)$$

Especificações:

- Se o usuário digitar um número negativo ou ímpar, o programa deve solicitar o valor de n novamente, até que o usuário digite n par e positivo.
- Cada uma das somas devem ser calculadas em uma função própria e os **resultados devem ser retornados** para o programa principal que os imprimirá.

BOA PROVA!