

Avaliação 1			Nota:
Curso Bacharelado em Ciência da Computação			
Disciplina Algoritmos e Programação 2			
Nome do(a) acadêmico(a)			
Nº de matrícula	Turma 2º Período	Data da Avaliação 11/12/2023	Professor(a) Ana Paula Freitas Vilela Boaventura
Tipo 1			

- A avaliação é **individual e sem consulta**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: Linguagem C, *String*, Registros, Módulos, Variáveis locais e Globais, Passagem de Parâmetros por valor;
- **O envio da prova será feito via e-mail oficial do discente para [ana\\_vilela@ufjf.edu.br](mailto:ana_vilela@ufjf.edu.br);**
- A interpretação do enunciado, faz parte da avaliação e qualquer tentativa de fraude acarretará abertura de processo administrativo na UFJ;
- Na avaliação, será levado em conta a lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza.

1 – (6.0 pontos) – Seja uma estrutura global do tipo *struct* cliente, que contenha os campos:

```
int CPF,
char nome [50];
float compras[3];
```

- Declare um vetor global com 3 posições do tipo *struct* cliente; (1.0 (um) ponto)
- No programa principal, invoque a função *le\_escreve( )*, para ler e escrever os campos do vetor de *struct*; (1.0 (um) ponto)
- No programa principal, invoque a função *verifica( )*, que confere se o nome “Maria” foi digitado. Se tiver sido digitado, o módulo *verifica* imprime na tela em qual posição do vetor o registro da Maria se encontra (1º, 2º ou 3º posição (2.0 (um) ponto)
- No programa principal, invoque a função *media( )*, que calcula a média dos valores comprados para cada cliente e imprima na tela. O subprograma *media( )* deverá informar o nome do cliente que efetuou o maior gasto (2.0 (dois) pontos)

2 – (4.0 pontos) – Faça um programa que calcule a expressão a seguir:

- Crie uma função principal que leia o valor de N, do tipo inteiro e de escopo local. Chame a função *soma(N)* e imprima na tela o resultado (1.0 ponto);
- Crie um arquivo *somatorio.h*, que tenha a função *int soma(int N)*, que seja capaz de calcular a seguinte expressão e retorne o valor do somatório (1.5 pontos por calcular o somatório corretamente e 1.5 para o Fibonacci):

$$\sum_{i=1}^N i + fib(i)$$

Biblioteca: *string.h*

*strcpy(str1, str2)*: faz a cópia do conteúdo da *str2* na *str1*

*strcmp(str1, str2)*: faz a comparação da *str1* com a *str2*

Avaliação 1		Nota:	
Curso <i>Bacharelado em Ciência da Computação</i>			
Disciplina <i>Algoritmos e Programação 2</i>			
Nome do(a) acadêmico(a)			
Nº de matrícula	Turma <i>2º Período</i>	Data da Avaliação <i>11/12/2023</i>	Professor(a) <i>Ana Paula Freitas Vilela Boaventura</i>
<b>Tipo 2</b>			

- A avaliação é **individual e sem consulta**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: Linguagem C, *String*, Registros, Módulos, Variáveis locais e Globais, Passagem de Parâmetros por valor;
- **O envio da prova será feito via e-mail oficial do discente para [ana\\_vilela@ufjf.edu.br](mailto:ana_vilela@ufjf.edu.br)**;
- A interpretação do enunciado, faz parte da avaliação e qualquer tentativa de fraude acarretará abertura de processo administrativo na UFJ;
- Na avaliação, será levado em conta a lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza.

1 – (6.0 pontos) – Seja uma estrutura global do tipo *struct* estudante, que contenha os campos:

```
int matricula, frequencia; //sendo a frequência o percentual.
char nome[50];
float nota[3];
```

- Declare um vetor global com 3 posições do tipo *struct* estudante; (1.0 (um) ponto)
- No programa principal, invoque a função *le\_escreve()*, para ler e escrever os campos do vetor de *struct*; (1.0 (um) ponto).
- No programa principal, invoque a função *Confere()*, que verifica se o nome “Aline” foi digitado. Se tiver sido digitado, o módulo verifica imprime na tela em qual posição do vetor o registro da Aline se encontra (1º, 2º ou 3º posição) (2.0 (um) ponto)
- No programa principal, invoque a função *Media()*, que verifica se o estudante foi aprovado ou não. Para ser aprovado, precisa ter uma média aritmética maior ou igual a seis e a frequência maior ou igual a 75%. (4.0 (dois) pontos)

2 – (4.0 pontos) – Faça um programa que calcule a expressão a seguir:

- Crie uma função principal que leia o valor de N, do tipo inteiro e de escopo local. Chame a função *soma(N)* e imprima na tela o resultado (1.0 ponto);
- Crie um arquivo *somatorio.h*, que tenha a função *int soma(int N)*, que seja capaz de calcular a seguinte expressão e retorne o valor do somatório (1.5 pontos por calcular o somatório corretamente e 1.5 para o Fatorial):

$$\sum_{i=1}^N i * i!$$

Biblioteca: *string.h*  
*strcpy(str1, str2)*: faz a cópia do conteúdo da *str2* na *str1*  
*strcmp(str1, str2)*: faz a comparação da *str1* com a *str2*

Avaliação 1			Nota:
Curso Bacharelado em Ciência da Computação			
Disciplina <b>Algoritmos e Programação 2</b>			
Nome do(a) acadêmico(a)			
Nº de matrícula	Turma 2º Período	Data da Avaliação 11/12/2023	Professor(a) Ana Paula Freitas Vilela Boaventura
<b>Tipo 3</b>			

- A avaliação é **individual e sem consulta**;

- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: Linguagem C, *String*, Registros, Módulos, Variáveis locais e Globais, Passagem de Parâmetros por valor;
- **O envio da prova será feito via e-mail oficial do discente para [ana\\_vilela@ufjf.edu.br](mailto:ana_vilela@ufjf.edu.br);**
- A interpretação do enunciado, faz parte da avaliação e qualquer tentativa de fraude acarretará abertura de processo administrativo na UFJ;
- Na avaliação, será levado em conta a lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza.

1 – (6.0 pontos) – Seja uma estrutura global do tipo *struct* paciente, que contenha os campos:

```
int CPF,
char nome [50];
float massaCorporal[3], altura; //massaCorporal é “quanto o paciente pesa”, em kg
```

- Declare um vetor global com 3 posições do tipo *struct* paciente; (1.0 (um) ponto)
- No programa principal, invoque a função *Le\_escreve()*, para ler e escrever os campos do vetor de *struct*; (1.0 (um) ponto)
- No programa principal, invoque a função *Atualiza()*, que altera o campo nome do 1º paciente para “Maria” e escreve na tela os nomes cadastrados. (1.0 (um) ponto)
- No programa principal, invoque a função *IMC()*, que calcula a média aritmética dos valores do vetor *massaCorporal*. Usando o peso médio, calcule e retorne o valor do IMC, sabendo que:  $IMC = \text{Peso} / (\text{altura})^2$ .

De uma forma generalizada e considerando apenas uma casa depois da vírgula, o módulo *IMC()* deverá informar se o peso dos pacientes está abaixo do peso abaixo (IMC menor ou igual a 18,4), normal (IMC entre 18,5 e 24,9) ou acima (maior ou igual a 25). (1.0 para calcular a média, 0.5 para calcular o IMC e 1.5 para indicar a saída correta no programa principal)

2 – (4.0 pontos) – Faça um programa que calcule a expressão a seguir:

- Crie uma função principal que leia o valor de N, do tipo inteiro e de escopo local. Chame a função *soma(N)* e imprima na tela o resultado (1.0 ponto);
- Crie um arquivo *somatorio.h*, que tenha a função *float soma(int N)*, que seja capaz de calcular a seguinte expressão e retorne o valor do somatório (1.5 pontos por calcular o somatório corretamente e 1.5 para o Fatorial):

$$S = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{N!}$$

Biblioteca: *string.h*

*strcpy(str1, str2)*: faz a cópia do conteúdo da *str2* na *str1*

*strcmp(str1, str2)*: faz a comparação da *str1* com a *str2*

Avaliação 1			Nota:
Curso Bacharelado em Ciência da Computação			
Disciplina Algoritmos e Programação 2			
Nome do(a) acadêmico(a)			
Nº de matrícula	Turma 2º Período	Data da Avaliação 11/12/2023	Professor(a) Ana Paula Freitas Vilela Boaventura
Tipo 4			

- A avaliação é **individual e sem consulta**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: Linguagem C, *String*, Registros, Módulos, Variáveis locais e Globais, Passagem de Parâmetros por valor;
- **O envio da prova será feito via e-mail oficial do discente para [ana\\_vilela@ufjf.edu.br](mailto:ana_vilela@ufjf.edu.br)**;
- A interpretação do enunciado, faz parte da avaliação e qualquer tentativa de fraude acarretará abertura de processo administrativo na UFJ;
- Na avaliação, será levado em conta a lógica, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza.

- 1 – (6.0 pontos) – Seja uma estrutura global do tipo *struct* produtos, que contenha os campos:
- ```
int identificador, quantidade [3]; /*sendo a quantidade em estoque para tamanho P, M,e G,
respectivamente.*/
char nome[50];
float preco;
```
- Declare um vetor global com 3 posições do tipo *struct* produtos; (1.0 (um) ponto)
  - No programa principal, invoque a função *le\_escreve( )*, para ler e escrever os campos do vetor de *struct*; (1.0 (um) ponto).
  - No programa principal, invoque a função *Busca( )*, que verifica se o campo cujo nome é “bermuda” consta nos registros. Se tiver, indique quantas peças no tamanho M, caso contrário informe que a peça não foi cadastrada (2.0 (um) ponto).
  - No programa principal, invoque a função *CalculaTotal( )*, que contabiliza o valor total de estoque, levando em conta o valor e a quantidade de cada peça. (2.0 (dois) pontos)
- 2 – (4.0 pontos) – Faça um programa que calcule a expressão a seguir:
- Crie uma função principal que leia o valor de N, do tipo inteiro e de escopo local. Chame a função *soma(N)* e imprima na tela o resultado (1.0 ponto);
  - Crie um arquivo *somatorio.h*, que tenha a função *int soma(int N)*, que seja capaz de calcular a seguinte expressão e retorne o valor do somatório (1.5 pontos por calcular o somatório corretamente e 1.5 para o Fibonacci):

$$S = 1 + \frac{Fib(1)}{1} + \frac{Fib(2)}{2} + \frac{Fib(3)}{3} + \dots + \frac{Fib(N)}{N}$$

Biblioteca: *string.h*  
*strcpy(str1,str2)*: faz a cópia do conteúdo da *str2* na *str1*  
*strcmp(str1,str2)*: faz a comparação da *str1* com a *str2*