

Curso <i>Bacharelado em Ciência da Computação</i>		Campus <i>Jataí</i>	
Disciplina <i>Algoritmos e Programação 2</i>			
Nome do(a) acadêmico(a)		Assinatura	
Nº de matrícula	Turma <i>2º Período</i>	Data da Avaliação <i>13/02/2023</i>	Professor(a) <i>Ana Paula Freitas Vilela Boaventura</i>
ATENÇÃO: Somente serão passíveis de REVISÃO avaliações resolvidas a TINTA. Tipo 1			

- A avaliação é **individual**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: *String*, Registros, Módulos, Variáveis locais e Globais;
- A interpretação do enunciado, faz parte da avaliação;
- 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 – (1.0 pontos) – Crie um menu interativo para selecionar os exercícios da prova e invoque as funções criadas nas bibliotecas *.h.

2 – (5.0 pontos) – Defina uma estrutura global do tipo *struct* estudante, que contenha os campos: matrícula, nome, média final. Em seguida, crie uma biblioteca *.h que faça as seguintes instruções:

- Declare um vetor com 3 posições do tipo *struct* cliente; (1.0 (um) ponto)
- Faça a leitura dos campos da *struct*; (1.0 (um) ponto)
- No terceiro elemento do vetor (terceiro registro), atualize o campo nome para “Francisco Pereira” (1.0 (um) ponto)
- Calcule e informe quantos alunos foram aprovados e reprovados, sabendo que para ser aprovado a média precisa ser maior do que 6.0 pontos. (2.0 (dois) pontos)

3 – (4.0 pontos) – Defina uma variável global do tipo inteiro N. Depois, crie uma biblioteca *.h que calcule o Fatorial de N, criando uma função int fatorial (). Não se esqueça de fazer o tratamento de erro para ler a variável N (apenas valor positivo e maior ou igual a 0).

Lembrando que:

$$1! = 1$$

$$N! = (N-1)! * \dots * 1!$$

Descrição de como serão avaliados:

- Fazer o tratamento de erro para N (1.0 (um) ponto)
- Criar a função que calcula o Fatorial. **Não será válido o uso de recursividade.** (2.0 (dois) pontos)
- Chamar a função corretamente no programa principal. (1.0 (um) ponto)

Curso <i>Bacharelado em Ciência da Computação</i>		Campus <i>Jataí</i>	
Disciplina <i>Algoritmos e Programação 2</i>			
Nome do(a) acadêmico(a)		Assinatura	
Nº de matrícula	Turma <i>2º Período</i>	Data da Avaliação <i>13/02/2023</i>	Professor(a) <i>Ana Paula Freitas Vilela Boaventura</i>
ATENÇÃO: Somente serão passíveis de REVISÃO avaliações resolvidas a TINTA. Tipo 2			

- A avaliação é **individual**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: *String*, Registros, Módulos, Variáveis locais e Globais;
- A interpretação do enunciado, faz parte da avaliação;
- 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 – 1.0 (um) ponto – Crie um menu interativo para selecionar os exercícios da prova e invoque as funções criadas nas bibliotecas *.h.

2 – 5.0 (cinco) pontos – Defina uma estrutura global do tipo *struct* cliente, que contenha os campos: nome, CPF e saldo. Em seguida, crie uma biblioteca *.h que faça as seguintes instruções:

- Declare um vetor com 3 posições do tipo *struct* cliente; 1.0 (um) ponto
- Faça a leitura dos campos da *struct*; 1.0 (um) ponto
- No primeiro elemento do vetor (primeiro registro), atualize o campo nome para “Manuel da Silva”; 1.0 (um) ponto
- Verifique o saldo e se for positivo, acrescente 10% de bônus sobre o valor que tem no saldo; 2.0 (dois) pontos

3 – 4.0 (quatro) pontos – Defina uma variável global do tipo inteiro N. Depois, crie uma biblioteca *.h que calcule o Fibonacci de N, criando uma função *int fibonacci ()*. Não se esqueça de fazer o tratamento de erro para ler a variável N (apenas valor positivo e maior ou igual a 0).

Lembrando que:

$$\begin{aligned}\text{Fib}(0) &= 1 \\ \text{Fib}(1) &= 1 \\ \text{Fib}(N) &= \text{Fib}(N-2) + \text{Fib}(N-1)\end{aligned}$$

Descrição de como serão avaliados:

- Fazer o tratamento de erro para N; 1.0 (um) ponto
- Criar a função que calcula o Fibonacci. **Não será válido o uso de recursividade.** 2.5 (dois vírgula cinco pontos)
- Chamar a função corretamente no programa principal. 0.5 (zero vírgula cinco ponto)

Curso <i>Bacharelado em Ciência da Computação</i>		Campus <i>Jataí</i>	
Disciplina <i>Algoritmos e Programação 2</i>			
Nome do(a) acadêmico(a)		Assinatura	
Nº de matrícula	Turma <i>2º Período</i>	Data da Avaliação <i>13/02/2023</i>	Professor(a) <i>Ana Paula Freitas Vilela Boaventura</i>
ATENÇÃO: Somente serão passíveis de REVISÃO avaliações resolvidas a TINTA. Tipo 3			

- A avaliação é **individual**;
- A avaliação tem o valor de 10,0 (dez) pontos;
- O conteúdo exigido compreende os seguintes tópicos apresentados no *Plano de Ensino* da disciplina: *String*, Registros, Módulos, Variáveis locais e Globais;
- A interpretação do enunciado, faz parte da avaliação;
- 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 – 1.0 (um) ponto – Crie um menu interativo para selecionar os exercícios da prova e invoque as funções criadas nas bibliotecas *.h.

2 – 4.0 (três) pontos – Declare três variáveis globais: a, b, c. Em seguida, crie uma biblioteca *.h que contenha um procedimento chamado void ORDENA () que ordena os seus valores do maior para o menor. O programa principal irá ler os valores, chamar a função ordena e em seguida imprimir os valores ordenados. Não será necessário fazer tratamento de erro para valores repetidos. Em outras palavras, se a entrada for: 4, 5, 4, a saída será 5, 4, 4.

- Entrada e saída no programa principal (1.0 (um) ponto)
- Criar a função void ORDENA (), que ordena os valores em ordem decrescente. (3.0 (dois) pontos).

3 – 5.0 (cinco) pontos – Defina uma estrutura global do tipo *struct* colaborador, que contenha os campos: nome, horas trabalhadas e valor que recebe por hora. Em seguida, crie uma biblioteca *.h que faça as seguintes instruções:

- Declare uma variável do tipo *struct* colaborador e faça a leitura dos campos; 1.0 (um) ponto
 - Crie a função float salário () que faça o cálculo do salário final, sabendo que: Salário Final = (horas trabalhadas * valor que recebe por hora) – imposto de renda. 4.0 (quatro) pontos
- O desconto de imposto de renda é definido conforme a Tabela 1.

Tabela 1: Percentual de imposto de renda incidido sobre a renda do trabalhador.

Renda	Imposto de renda
De 0,00 a R\$ 2.000,00	Isento
De R\$ 2.000,01 até R\$ 3.000,00	8%
De R\$ 3.000,01 até R\$ 4.500,00	18%
Acima de R\$ 4.500,01	28%

Não é preciso fazer o cálculo de imposto escalonado. Assim, se o indivíduo receber R\$ 5.000,00 o programa irá descontar 28%, fica, portanto, R\$1.400,00 o valor de desconto.