

Prova 1

Algoritmos e Estruturas de Dados I - turma TE

Professor: Pedro O.S. Vaz de Melo

25 de março de 2014

Nome: _____
escrevendo o meu nome eu juro que seguirei o código de honra

Código de Honra para este exame:

- Não darei ajuda a outros colegas durante os exames, nem lhes pedirei ajuda;
- não copiarei nem deixarei que um colega copie de mim;
- não usarei no exame elementos de consulta não autorizados.

Informações importantes:

- Em questões que pede um **programa**, este deve ser completo, com bibliotecas (incluindo, quando necessário, a biblioteca `prova1.h`), função `main`, etc. Se deve ser feita uma **função**, somente a função é suficiente. Se deve ser feito um **procedimento**, somente o procedimento é suficiente.
- A interpretação das questões da prova faz parte do critério de avaliação. Caso tenha dúvida sobre a sua interpretação de uma determinada questão, escreva as suas suposições na resolução da mesma.
- As funções implementadas no módulo `prova1.h` podem ser usadas em **qualquer** exercício da prova. Além disso, se você usar uma função do módulo `prova1.h`, considere que ela está implementada de forma correta.

Referências:

Função/Operador	Descrição	Biblioteca	Exemplo
<code>float exp(float x)</code>	retorna e^x	<code>math.h</code>	<code>exp(1)</code> retorna $e^1 = 2.71828$
<code>float pow(float b, float e)</code>	retorna b^e	<code>math.h</code>	<code>pow(2,3)</code> retorna $2^3 = 8$
<code>int abs(int x)</code>	retorna $ x $	<code>stdlib.h</code>	<code>abs(-3)</code> retorna $ 3 = 3$
<code>double sqrt(double x)</code>	retorna \sqrt{x}	<code>math.h</code>	<code>sqrt(9)</code> retorna $\sqrt{9} = 3$
<code>%</code>	retorna o resto da divisão	-	<code>20 % 3</code> retorna 2

1. (14 points) Para as questões a seguir, considere que as implementações serão feitas no módulo "`prova1.h`".

a. (3 pts) Um matemático maluco lhe procurou pois precisa de uma implementação em C da função:

$$f(x, \alpha) = \begin{cases} \alpha\sqrt{-x} & \text{se } x < 0 \\ \alpha e^{-\alpha x} & \text{se } x \geq 0 \text{ e } x < |\alpha^3| \\ \alpha & \text{se } x \geq |\alpha^3| \end{cases} \quad (1)$$

Assim, implemente uma **função** de nome `frouca` que recebe os parâmetros x e α e retorna $f(x, \alpha)$, todos pontos flutuantes.

b. (4 pts) Escreva uma **função** de nome `numeroDivisores` que retorna o número de divisores comuns entre dois números inteiros x e y que são diferentes de 1. Assim, a função `numeroDivisores` deve retornar 3 se os parâmetros de entrada forem 18 e 12 (divisores 2, 3 e 6) e 0 se os parâmetros de entrada forem 13 e 7. Essa função deve ter o seguinte protótipo:

```
int numeroDivisores(unsigned int x, unsigned int y);
```

c. (3 pts) Implemente uma função que recebe dois pontos flutuantes x e y e que retorna o menor inteiro que seja maior que ambos. O protótipo da função é:

```
int menorIntMaior(float x, float y);
```

d. (4 pts) Escreva um **procedimento** de nome `aumentaSeTemDivisores` que recebe como parâmetro dois endereços de memória de variáveis inteiras `end_var1` e `end_var2`. A função deve verificar se o conteúdo desses endereços de memória têm um divisor comum diferente de 1. Caso negativo, a função deve armazenar o valor 0 nos dois endereços de memória. Caso positivo, a função deve multiplicar os dois valores e armazenar o resultado em ambos endereços de memória.

2. (5 points) Escreva um **programa** que lê três números inteiros do teclado: x_{min} , x_{max} e α e imprime na tela os valores resultantes da função `frouca` com parâmetro α para todos os valores inteiros entre x_{min} e x_{max} (inclusive para ambos). Enquanto o usuário inserir um valor de x_{min} maior ou igual a x_{max} , o programa deve pedir a ele novos valores de x_{min} e x_{max} .

3. (3 points)

a. (2 pts) Complete o código abaixo, considerando que as variáveis x e y vão ser usadas como parâmetros nas linhas 7, 8, 9 e 11:

```
1: #include <stdio.h>
2: #include _____
3:
4: void main(void) {
5:     int x,y;
6:     printf("Digite os valores de x e y\n");
7:     scanf(_____);
8:     aumentaSeTemDivisores(_____);
9:     printf("Os novos valores de x e y sao: x=_____ e y=_____\n", _____, _____);
10:    aumentaSeTemDivisores(_____);
11:    printf("Os novos valores de x e y sao: x=_____ e y=_____\n", _____, _____);
12: }
```

b. (1 pt) O que foi impresso nas linhas 9 e 11 caso o usuário tenha entrado com os valores $x = 4$ e $y = 6$ na linha 7?