

Prova 3

Algoritmos e Estruturas de Dados I - turma W

Professor: Pedro O.S. Vaz de Melo

18 de junho de 2013 (valor: 20 pontos)

Nome:

escrevendo o meu nome eu juro que seguirei o código de honra

Código de Honra para este exame (baseado no *Honor Code* da Universidade de Stanford):

- 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:

- Considere que todos os procedimentos e funções pedidas nesta prova serão implementados no módulo `prova3.h`.
- Em questões que pede um **programa**, este deve ser completo, com bibliotecas (incluindo o módulo `prova3.h` quando necessário), 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.

Referências:

Função/Operador	Biblioteca	Exemplo
<code>fscanf(FILE *stream, const char *format, endereço das variáveis);</code>	<code>stdio.h</code>	<code>fscanf(arq, "%f", &nota1);</code>
<code>fprintf(FILE *stream, const char *format, valores/variáveis);</code>	<code>stdio.h</code>	<code>fprintf(arq, "%d", aux);</code>

1. (5 points) Escreva uma função RECURSIVA de protótipo `int maiorDivPotencia2(int n)` que retorna a maior potência de 2 que seja divisor de um inteiro n . Um número n é uma potência de 2 se pode ser representado por $2^k, k \geq 0$. Exemplo: `maiorDivMult2(56)=8`, pois $56 = 2 * 2 * 2 * 7 = 8 * 7$ e $8 = 2^3$.

2. (5 points) Escreva um procedimento para ordenar de forma crescente um vetor de inteiros de n posições. Protótipo: `void ordena(int v[], int n);`

3. (5 points) Escreva um procedimento de nome `aloca` para criar um vetor de inteiros por alocação dinâmica de memória e atribuir 0 a todas as suas posições. Esse procedimento recebe dois parâmetros: um ponteiro para inteiro v **por referência** e um inteiro n por valor. Ele deve alocar dinamicamente n posições de memória para armazenar inteiros e atribuir a referência das mesmas a v . Feito isso, deve-se atribuir o valor 0 a todas as suas n posições.

4. (5 points) Escreva um programa para ler um arquivo `vetor.txt` contendo um vetor de n posições. Considere que o primeiro campo desse arquivo contém o valor de n e os demais campos contém os valores do vetor. Ex: se o conteúdo do arquivo for 5 12 100 1 4 78, então 5 é o tamanho do vetor{12, 100, 1, 4, 78}. Ordene esse vetor e o salve em um arquivo de nome `vetor_ordenado.txt` usando o mesmo formato do arquivo `vetor.txt`, isto é, o primeiro campo deve conter n e os demais campos devem conter os valores do vetor ordenado. Lembrando que você pode usar qualquer função/procedimento que foi pedido para você implementar nesta prova.

5. (5 points) (EXTRA) Considere o código do programa que usa a biblioteca Allegro dado em anexo e responda (a figura `Angry-Bird.jpg` é de um pássaro raivoso vermelho como o do jogo *Angry Birds*):

a. (1 pt) O que o programa mostra na tela durante a sua execução?

b. (2 pts) Como eu faço para fazer que a figura `Angry-Bird.jpg` faça uma trajetória qualquer de projétil na tela (semelhante ao TP1)? Indique as linhas e o código que você modificaria/acrescentaria. Exemplos de resposta válida: “entre as linhas 19 e 20 eu acrescentaria o código: `float velocidade = 100;` e a linha 36 eu alteraria para: `redraw = false;`”.

c. (2 pts) Como eu faço para colocar um alvo qualquer na tela e indicar se o projétil acertou o alvo? Indique as linhas e o código que você modificaria/acrescentaria como no exercício anterior.