

Prova 3

Algoritmos e Estruturas de Dados I

Professor: Jefersson dos Santos e Pedro Vaz de Melo

03 de janeiro de 2017 (valor: 30 pontos)

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.

1. (6 points) Escreva uma função RECURSIVA de protótipo `int ehPotencia2(int n)` que retorna 1 se n for uma potência de 2 ou 0 caso contrário. Um número n é uma potência de 2 se pode ser representado por 2^k , $k \geq 0$. Exemplo: `ehPotencia2(16)` retorna 1, pois $16 = 2 * 2 * 2 * 2 = 2^4$.

2. (6 points)

Escreva um procedimento RECURSIVO de protótipo `void imp2Cont(int i, int n)` que imprime a contagem crescente e, logo depois, a contagem decrescente de i até n . Sua função não pode usar *loops* (`for`, `while`, etc). Exemplo: `imp2Cont(2,5)` imprime

```
2
3
4
5
4
3
2
```

3. (8 points) Ao preencher formulários online, muitas vezes os usuários colocam diversos espaços em branco antes e depois do texto digitado. Apesar de parecer insignificante, isso pode gerar problemas em diversos processos como, por exemplo, de comparação de *strings* e de alocação de memória.

Assim, escreva um procedimento de nome `trim` que recebe uma *string* `str` como parâmetro **por referência** e a modifica removendo todos os espaços em branco que a precede e a sucede, caso necessário. Em suma, você deve transferir a *string* sem os espaços em branco iniciais e finais para uma nova área de memória (*heap*) e, após isso, fazer com que o parâmetro `str` aponte para essa nova área. Exemplo: se a *string* inicial for " AEDS1 ", você a deve transformar para "AEDS1". Não se esqueça de desalocar o espaço da *string* antiga e usar o terminador `\0`. A função deve ter o seguinte protótipo:

```
void trim(char **str);
```

4. (10 points) Uma empresa está desenvolvendo um sistema de cadastro de pessoas e alguns métodos estão faltando. Você foi convocado para implementar os métodos `retornaIdade` e `cidadeMaisNovo`, descritos abaixo. Os tipos de dados que você deve considerar são:

- (a) Um **tipo de dados** `Data` para representar uma data, com o seu ano, mês e dia.
- (b) Um **tipo de dados** `Endereco` para representar um endereço, contendo todas as informações necessárias para que uma carta seja enviada para o endereço correspondente.
- (c) Um **tipo de dados** `Pessoa` para representar uma pessoa, contendo informações como nome, idade, data de nascimento e endereço.

d. (5 pts) Implemente uma função de nome `retornaIdade` que recebe uma `Data` como parâmetro e retorna quantos anos se passaram desde essa data até o dia de hoje. Nesta questão você deve utilizar a função `Data hoje()`, que retorna uma estrutura `Data` com os seus campos automaticamente preenchidos com valores correspondentes a data atual. Exemplo: se a função `hoje()` retornar 02/01/2017 e se a data passada como parâmetro for 31/12/2015, a sua função deve retornar 1.

```
int retornaIdade(struct Data nascimento);
```

e. (5 pts) Implemente um procedimento que recebe um vetor de pessoas e o seu tamanho n como parâmetros e **imprime** a cidade da pessoa mais nova. O seu procedimento deve ter o seguinte protótipo:

```
void cidadeMaisNovo(Pessoa p[], int n);
```

```
typedef struct Data {  
    int ano;  
    int mes;  
    int dia;  
} Data;
```

```
typedef struct Endereco {  
    char rua[200];  
    char cep[20];  
    int numero;  
    char complemento[20];  
    char cidade[200];  
    char estado[100];  
    char pais[100];  
} Endereco;
```

```
typedef struct Pessoa {  
    char nome[200];  
    Data nasc;  
    int idade;  
    Endereco end;  
} Pessoa;
```

```
Data hoje();
```