

Programação Imperativa – EI (1º ano)

Exame de Época Especial

15 de Setembro de 2011 (14h00)

Dispõe de **2:00 horas** para realizar este teste.

Questão 1 (cálculo numérico, 4v)

Codifique em C uma função que recebe dois argumentos do tipo inteiro, a e b , e imprime o somatório dos números primos existentes no intervalo fechado $[a, b]$.

Apresente definições para duas versões desta função:

- `int somait(int a, int b)` — que realiza o cálculo de forma iterativa;
- `int somarec(int a, int b)` — que realiza o cálculo de forma recursiva.

Questão 2.1 (arrays 2v)

Especifique em C uma função "numcomuns" que recebe quatro argumentos: dois arrays de inteiros a e b e o número de elementos em cada um dos arrays $size_a$ e $size_b$ e devolve um inteiro que corresponde ao número de elementos que são comuns aos dois arrays. Considere cada array como uma lista de elementos sem repetições.

```
int numcomuns( int a[], int b[], int size_a, int size_b )
{
    ...
}
```

Questão 2.2 (strings 2v)

Especifique em C a função `strcspn(s1,s2)` que calcula o comprimento do segmento inicial da string $s1$ que não contém nenhum dos caracteres presentes em $s2$. Por exemplo:

```
... i = strcspn("Programacao Imperativa", "AEIOU");
    i ficaria com o valor 12 que é o índice de 'I'
```

```
int strcspn( char *s1, char *s2 )
{
    ...
}
```

Questão 3 (ordenação, 4v)

Considere a seguinte estrutura de dados que pretende modelar uma lista de alunos:

```
typedef struct sAluno
{
    char *numero;
    char *nome;
    char *curso;
} Aluno;

typedef struct sLLAluno
{
    Aluno a;
    struct sLLAluno *seg;
} *LLAluno, NodoLLAluno;
```

Especifique em C uma função que recebe uma lista de alunos (do tipo `LLAluno`) e a devolve com os elementos ordenados por ordem crescente e alfabética de nome.

Questão 4 (modelação de estruturas de dados, 4v)

Pretende-se especificar um novo sistema de informação que permita armazenar a informação e gerir as várias lojas das várias empresas instaladas num centro comercial.

Uma empresa é caracterizada pela sua designação, pela morada da sua sede social e pela lista de lojas que detem no centro comercial (cada vez é mais normal encontrarmos várias lojas da mesma empresa no mesmo centro comercial).

Uma loja é caracterizada pelo seu nome, a localização (em termos de andar, sector, e número de porta), o ramo comercial (cada loja pode conter uma lista de ramos: vestuário, ourivesaria, livraria, papelaria, etc.) e um slogan publicitário (um texto livre, mas curto). Por sua vez, um ramo é apenas uma string.

Atendendo a esta descrição desenvolva as seguintes alíneas:

- Especifique em C os tipos de dados capazes de suportar o sistema de informação descrito;
- Codifique em C uma função ??? `distPorRamos(CComercial c)` que recebe o sistema de informação dum centro comercial e produz (defina o tipo do resultado) uma lista dos ramos existentes no centro e que indique para cada ramo quantas lojas existem.

Questão 5 (estruturas de dados, 4v)

Considere a estrutura de dados descrita pelos tipos abaixo declarados, que servem para suportar a Agenda de 1 mês num telemóvel Nokia:

```
typedef struct sTarefa
{
    int dia;
    int hora;
    char *tipo;
    char *descricao;
} Tarefa;

typedef struct sLLTarefa
{
    Tarefa t;
    struct sLLTarefa *seg;
} *LLTarefa, NodoLLTarefa;

typedef struct sAgendaMensal
{
    int mes;
    LLTarefa ag[31];
} AgendaMensal;
```

Com base nesta especificação desenvolva as seguintes alíneas:

- Codifique em C uma função `Tarefa leTarefa()` para ler os dados de uma tarefa (dia, hora, tipo, descrição) e criar uma célula com essa informação;
- Codifique em C uma função `AgendaMensal pushTarefa(AgendaMensal a, Tarefa t)` para inserir essa célula na cabeça da lista de tarefas desse dia;
- Codifique em C uma função `AgendaMensal sortedTarefas(AgendaMensal a, Tarefa t)` para inserir essa célula ordenadamente na lista de tarefas desse dia (ordem crescente de hora, sem repetições à mesma hora);
- Codifique em C uma função `void listarTarefas(AgendaMensal a, int dia)` para imprimir todas as tarefas agendadas para 1 dia (recebido como argumento).