

Programação Imperativa – CC (1º ano)

Teste de Avaliação Final

Data: 27 de Junho de 2008

Hora: 9:00

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

Responda na folha fornecida em anexo e não se esqueça de preencher o cabeçalho

Questão 1 (cálculo numérico)

Defina uma **função** C que, dados os valores a , b e c , calcule a maior raiz de $ax^2 + bx + c = 0$, ou seja

$$x = \max\left(\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}\right)$$

Questão 2 (arrays)

Uma das áreas da tradução com maior impacto nos últimos anos é a tradução de aplicações (software). Esta tarefa não se limita a traduzir as mensagens da aplicação, mas também adaptar as moedas, medidas e outras strings que são específicas de determinados países. A estes diferentes tipos de adaptação dão-se os nomes de *internationalization*, *localization* e *globalization*. Dado o grande tamanho destas palavras é habitual abreviarem-se por *i18n*, *l10n* e *g11n*.

Desta ideia surgiu uma nova forma de codificar texto: pegar em todas as palavras com mais de três caracteres, e substituí-las pelo primeiro carácter, seguido do número de caracteres da palavra sem contar a primeira e última letra, e o último carácter da palavra.

Desenvolva uma função C que recebe uma string (terminada em '\0') e a retorna de forma compactada (considere que o texto contém apenas letras minúsculas e espaços). Por exemplo:

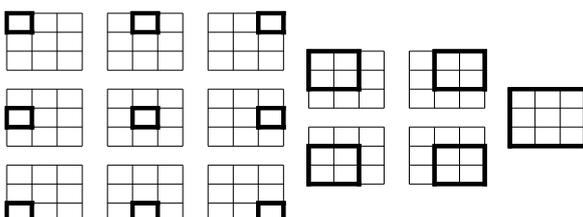
`mais vale só que mal acompanhado`

deverá ser substituída por:

`m2s v2e só que mal a10o`

Questão 3 (recursividade)

Uma das tarefas habituais para passar tempo na casa-de-banho consiste em tentar contar o número de quadrados existentes num quadrado dividido em n pequenos quadrados. A figura que se segue mostra como esse número se obtém:

$$f(n) = \sum_1^n n^2$$


Implemente duas funções que calculem f : uma iterativa e uma outra recursiva.

Questão 4 (estruturas dinâmicas)

Considere a seguinte definição em C que especifica os tipos de dados e a assinatura de algumas funções que implementam uma árvore binária de procura de inteiros:

```
typedef struct sAbin {
    int valor;
    struct sAbin *esq, *dir;
} Nodo, *Abin;

Abin insere( Abin a, int n );
void lista( Abin a );
```

Com base nesta especificação responda às seguintes alíneas:

a) Desenhe a árvore binária que se obtém depois de realizar as seguintes operações de inserção:

```
...
Abin a1;
...
a1 = insere(insere(insere(insere(a1,17),23),8),9);
a1 = insere(insere(insere(a1,21),5),42);
...
```

b) Dada uma árvore binária, a cada elemento pode ser associado o seu "*caminho*", que não é mais do que uma lista de Booleanos. Por exemplo, a lista [False, True, False] corresponderia ao caminho de um nodo que está à direita do nodo raiz, à esquerda desse segundo nodo e à direita do terceiro nodo. O valor False indica que o elemento está à direita, o valor True que está à esquerda, e a lista vazia que o elemento está na raiz. Considere a seguinte definição de lista para suportar os caminhos:

```
typedef int boolean;
#define TRUE 1
#define FALSE 0

typedef struct sCaminho {
    boolean valor;
    struct sCaminho *seg;
} Elem, *Caminho;
```

Defina uma função com a seguinte assinatura `boolean cvalido(Abin a, Caminho c)`; que, dada uma árvore e um caminho, diz se esse caminho é válido nessa árvore, i.e., se corresponde a algum nodo da árvore (se o caminho no final não atingir um nodo válido a função retornará o valor falso).

- c) Defina uma outra função `int selecciona(Abin a, Caminho c)`; que, dada uma árvore e um caminho válido nessa árvore, retorna o elemento identificado por esse caminho.
- d) Defina uma função `Caminho procura(Abin a, int n)`; que, dada uma árvore e um elemento, procura esse elemento na árvore. Em caso de sucesso, a função deverá retornar o caminho desse elemento, em caso de insucesso deve retornar um caminho vazio.

Questão 5 (a função X)

Considere as definições apresentadas na questão anterior relativas a árvores binárias de procura (ABP) de inteiros.

Codifique em C uma função de nome X que realiza uma travessia por níveis da ABP (primeiro escreve os nodos com peso 1, depois os nodos com peso 2 e por aí adiante) escrevendo o valor de cada nodo no monitor.