

Programação – Exame da Época Normal

17 de junho de 2024 – Duração: 120 minutos

LEI, LEI-PL, LEI-CE

1. Considere as seguintes definições:

```
typedef struct a autor;  
typedef struct b livro;  
  
struct a{  
    char nome[100];    // Nome do autor  
    int idA;           // Identificador numérico do autor  
    int nLivros;       // Número de títulos no catálogo da livraria  
};  
  
struct b{  
    char titulo[100], isbn[20];    // Título e isbn do livro  
    int idA;                       // Identificador do autor  
    float preco;                   // Preço do livro  
};
```

Uma livraria armazena num ficheiro binário informação sobre os autores que comercializa e os livros que tem disponíveis. A organização do ficheiro binário é a seguinte:

- No início encontra-se um inteiro X indicando quantos autores comercializa.
- Seguem-se X estruturas do tipo *autor*, cada uma delas com dados de um dos autores comercializados.
- A seguir encontra-se outro inteiro Y, indicando quantos livros diferentes são comercializados.
- Finalmente, o ficheiro contém Y estruturas do tipo *livro*, uma para cada um dos títulos comercializados. Pode assumir que só são comercializados livros de autores identificados numa das estruturas do tipo *autor*. A figura ilustra um exemplo com 2 autores e 4 livros. A informação no ficheiro não está ordenada por nenhum critério em particular.

Escreva uma função em C que efetue as seguintes operações:

- Atualize o ficheiro binário, aumentando em 10% o preço dos livros do autor com mais títulos na livraria. Caso exista empate entre vários autores com número máximo de livros, não deve ser efetuada nenhuma atualização de preço.
- Crie um ficheiro de texto com informação dos títulos existentes de um determinado autor. A informação de cada um dos livros deve surgir numa linha com o seguinte formato: ISBN - Título, Nome do Autor. Caso o autor indicado não esteja registado no ficheiro binário, deve ser escrito “Autor Inexistente” no ficheiro de texto.

A função recebe como parâmetros os nomes dos ficheiros e o identificador numérico do autor a considerar para criar o ficheiro de texto. Devolve 1 se tudo correr bem, ou 0, no caso de existir algum problema no acesso aos ficheiros.

2	
Artur Pires	12 3
Carlos Vaz	37 1
4	
A Casa	3X5RT 12 18.9
O Gato	DFG544 12 25.0
O Pato	ASD23 37 9.1
O Jardim	PDF12 12 31.5

[Cotação: 30%]

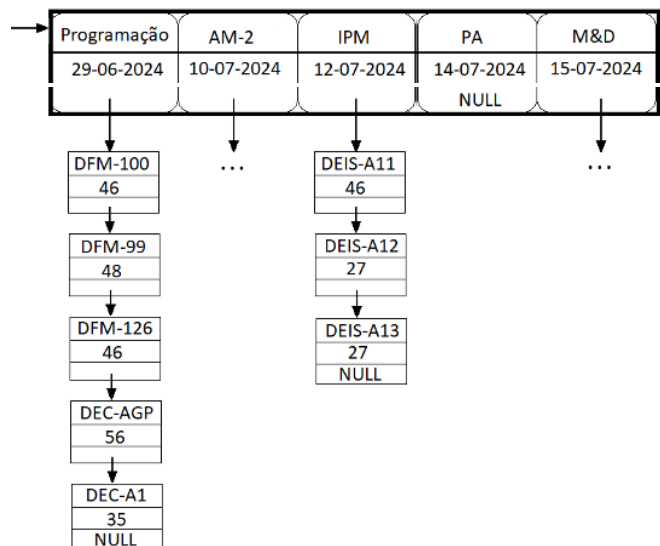
2. Considere as seguintes definições, utilizadas para criar uma estrutura dinâmica que armazena informação sobre a reserva de salas para os exames da época normal da LEI:

```
typedef struct dadosUc    uc, *pUc;
typedef struct dadosSala  sala, *pSala;
typedef struct {int dia, mes, ano;} data;

struct dadosUc{
    char refUc[100];    // Nome da Unidade Curricular
    data dataEx;        // Data do exame
    pSala lista;        // Ponteiro para a lista de salas
};

struct dadosSala{
    char refSala[100];  // Nome da sala
    int capacidade;     // Capacidade de sala
    pSala prox;
};
```

A estrutura dinâmica tem as seguintes características: uma tabela dinâmica é constituída por estruturas do tipo *uc* (uma para cada unidade curricular). As estruturas na tabela estão ordenadas por data de realização do exame. Pode assumir que não há vários exames marcados para o mesmo dia. A cada unidade curricular está associada uma lista ligada simples (através do campo *lista*), constituída por nós do tipo *sala*. Cada lista contém informação das salas reservadas para o exame em questão. Caso ainda não exista nenhuma sala reservada para uma determinada unidade curricular, o campo *lista* tem o valor NULL. Ao lado pode consultar um exemplo da estrutura dinâmica, em que o exame de Programação tem 5 salas reservadas.



Escreva uma função em C que apresente na consola o nome de todas as unidades curriculares cujo exame tem reservada uma determinada sala. A função recebe, como parâmetros, o endereço do vetor dinâmico e a sua dimensão, assim como o nome da sala a pesquisar.

[Cotação: 20%]

3. Escreva uma função em C que elimine da estrutura dinâmica todas as unidades curriculares cujo exame decorra antes de uma determinada data. A função deve eliminar as unidades curriculares do vetor dinâmico, libertando as respetivas listas ligadas com a reserva das salas. O cabeçalho da função é o seguinte:

```
pUc elimina(pUc v, int *nUcs, data d);
```

A função recebe, como parâmetros, o endereço do vetor dinâmico, o endereço de uma variável inteira onde se encontra a dimensão do vetor e a data a considerar. Devolve o endereço do vetor dinâmico depois da atualização. Em caso de erro de gestão de memória, a função devolve a estrutura dinâmica original.

[Cotação: 40%]

Programação – Exame da Época Normal

17 de junho de 2024 – Duração: 120 minutos

LEI, LEI-PL, LEI-CE

Nome: _____ Número: _____

4. Funções Recursivas

Qual é o output na consola quando é executado o seguinte programa em C?

```
#include <stdio.h>

int f(int x){
    if(x<=1)
        return x;
    else if(x%2 == 0){
        printf("%d\t", x);
        return f(x/2);
    }
    else
        return x + f(x / 3);
}

int main() {
    printf("%d\n", f(13));
    return 0;
}
```

Resposta:

4 2 _____

4 2 13 _____

13 _____

4 2 14 _____

4 2 1 _____

[Cotação: 10%] (uma resposta errada não desconta)