

Programação – Época Especial de Setembro

Eng^a Informática; Eng^a Informática – Pós-laboral; Eng^a Informática - Curso Europeu

Duração: 2h30m

03/09/2015

Atenção: É obrigatório apresentar uma estratégia genérica para cada um dos exercícios.

1. Um banco armazena num **ficheiro binário** os movimentos efetuados ao longo de uma semana pelos seus clientes. A informação associada a cada movimento (nome do cliente e montante) é armazenada numa estrutura do tipo:

```
struct lev{
    char nome[100];
    int montante;
};
```

Se o valor do campo montante for positivo trata-se de um depósito. Caso contrário refere-se a um levantamento.

As estruturas estão organizadas no ficheiro pela ordem em que os movimentos são efetuados. Na figura pode ser consultado um exemplo de um ficheiro binário. Como se pode ver, ao longo da semana a Sandra Pinto só efetuou 1 depósito, enquanto que a Ana Silva realizou 1 depósito e 2 levantamentos. Ao responder a esta questão assuma que não existem clientes com nomes iguais.

Jorge Sousa
50
Ana Silva
-30
Carlos Vaz
-10
Sandra Pinto
50
Ana Silva
-50
Carlos Vaz
25
Ana Silva
200
Jorge Sousa
100

Desenvolva uma função em C que crie um **ficheiro de texto** contendo um resumo detalhado dos movimentos realizados por cada cliente. Cada linha deste ficheiro deve conter a informação de cada cliente no seguinte formato:

Nome: valor total movimentado (valor total dos depósitos: valor total dos levantamentos)

A seguir ilustra-se como deveria ficar o ficheiro de texto com base na informação do exemplo:

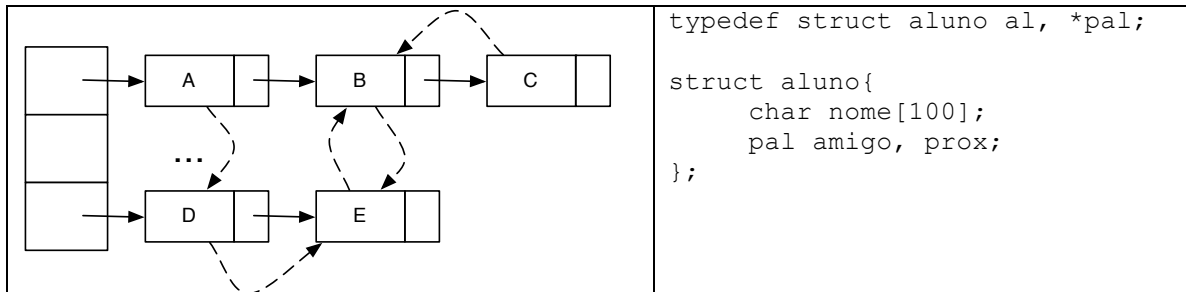
```
Jorge Sousa: 150 (150 : 0)
Ana Silva: 120 (200 : -80)
Carlos Vaz: 15 (25 : -10)
Sandra Pinto: 50 (50 : 0)
```

A função recebe como argumento os nomes dos ficheiros a processar.

2. Desenvolva uma **função recursiva em C** que some todos os valores pares armazenados numa lista ligada simples constituída por elementos do tipo *struct no*. A função recebe um ponteiro para o início da lista e devolve o valor calculado.

```
struct no{
    int val;
    struct no *prox;
};
```

3. Os alunos pertencentes às várias turmas de uma escola estão armazenados numa estrutura dinâmica com o seguinte formato:



Existe um vetor de ponteiros do tipo *pal*. Cada elemento deste vetor é um ponteiro para o início da lista de alunos de uma determinada turma (existe uma lista de alunos por turma). Cada um dos nós das listas de alunos é do tipo *al*. O campo *amigo* de cada aluno aponta para o seu melhor amigo (as setas a tracejado na figura indicam quais os melhores amigos de cada um dos alunos representados).

a) Desenvolva uma função em C que calcule quantos alunos da escola têm o seu melhor amigo na turma a que pertencem. No exemplo em cima, o aluno C tem o seu melhor amigo na mesma turma, enquanto que o aluno A tem o seu melhor amigo numa turma diferente. A função recebe o vetor de ponteiros e o número de turmas da escola como argumentos. Devolve como resultado o valor contabilizado.

b) Desenvolva uma função em C que construa uma lista ligada contendo os alunos que têm exatamente uma pessoa a considerá-los como melhores amigos. Considerando o exemplo, a lista deveria incluir apenas o aluno D. A função recebe o vector de ponteiros e o número de turmas da escola como argumentos. Devolve como resultado um ponteiro para o início da lista que for criada. Esta lista deve ser constituída por nós do tipo *al* (o campo *amigo* deve ficar a NULL), **ordenados alfabeticamente** pelo nome do aluno. Todo o espaço necessário para criar a lista resumo deve ser requisitado pela função (a estrutura dinâmica original não deve ser modificada).