

## Centro de Ciências Exatas, Arquitetura e Engenharia

Professor:	Ciro Cirne Trindade
Disciplina:	Estruturas de Dados-I
Cursos:	Ciência da Computação/Sistemas de Informação
Data de entrega:	18/05/2021
Grupo:	Até 4 alunos

### 1º Trabalho Prático – Quadro de Medalhas

Neste ano serão realizadas as Olimpíadas 2020, que foram adiadas devido a pandemia de Covid-16. O quadro de medalhas dos países participantes será exibido através de um placar eletrônico de última geração. Você foi convidado para viajar até Tóquio no Japão, aonde será a sede destes jogos, para, a partir de uma relação das provas e os países vencedores, desenvolver o sistema para este placar.

Serão quase trezentos países participantes e até 1000 modalidades de provas, no máximo.

#### Entrada

A entrada contém um único caso de teste que consiste em uma relação com o resultado de todas as provas realizadas nas Olimpíadas de Tóquio. Cada prova ou modalidade contém quatro linhas de informação: a primeira linha contém a descrição da prova, com, no máximo, 50 caracteres; a segunda linha contém o país que ficou campeão nesta modalidade; a terceira linha contém o país vice-campeão e; a última linha contém o país que ficou com a medalha de bronze na referida prova. Os nomes dos países têm, no máximo, 30 caracteres. O final da entrada é determinado por EOF.

#### Saída

Como saída, deve ser impresso o quadro de medalhas das Olimpíadas de Tóquio. A primeira linha contém a informação “Quadro de Medalhas”. Cada uma das próximas linhas conterá o nome de um país seguido pelo respectivo número de medalhas de ouro, prata e bronze que este país conquistou, separadas por um espaço em branco. O critério de desempate é, na ordem, o número de medalhas de ouro seguido pelo número de medalhas de prata e de bronze. Se países empatarem nestes três critérios, a listagem será por ordem ascendente do nome do país participante.

#### Exemplo

Entrada	Saída
Arremesso de disco Finlandia Noruega	Quadro de Medalhas Finlandia 2 0 1 Estados Unidos 1 1 0

Sri Lanca	Russia 1 0 1
Levantamento de peso	Islandia 1 0 0
Russia	Estonia 0 1 1
Africa do Sul	Africa do Sul 0 1 0
Estonia	Canada 0 1 0
Natacao 100 metros	Noruega 0 1 0
Estados Unidos	Sri Lanca 0 0 2
Canada	
Sri Lanca	
Hipismo	
Finlandia	
Estados Unidos	
Finlandia	
Handebol	
Islandia	
Estonia	
Russia	

## Requisitos Obrigatórios

Para a solução deste problema deve-se usar obrigatoriamente uma lista representada de forma sequencial, cujo arquivo de cabeçalho que define o TAD lista e os protótipos de suas operações é mostrado a seguir.

```
/* lista.h
 * Definição do TAD lista e protótipos de suas operações. */

#ifndef _LISTA_H
#define _LISTA_H

#include <stdbool.h>

#define LST_MAX 300

typedef struct {
    char pais[31];
    int ouro;
    int prata;
    int bronze;
} lst_info;

typedef struct {
    lst_info itens[LST_MAX];
    int n;
} lista;

/* função que inicializa a lista */
void lst_iniciar(lista *);

/* função que insere um valor no final da lista ou atualiza esse
```

```

valor na lista */
bool lst_inserir(lista *, lst_info);

/* função de devolve a posição de um país na lista ou -1 */
int lst_procurar(lista, const char *);

/* compara 2 valores da lista, devolve 0 se forem iguais, < 0 se
 * o 1o. for menor que o 2o., > 0 se o 1o. for maior que o 2o. */
int lst_compara(lst_info, lst_info);

/* função que ordena a lista por Heapsort */
void lst_ordenar(lista *);

/* função que imprime o quadro de medalhas */
void lst_imprimir(lista);

#endif // _LISTA_H

```

Após a entrada de dados a lista deve ser ordenada através da função `lst_ordenar()`, que deve usar obrigatoriamente o algoritmo de ordenação Heapsort.

A função `lst_compara()` compara dois elementos da lista e seu valor de retorno está descrito nos comentários dessa função. Sugere-se usar essa função para comparar os elementos da lista no algoritmo de ordenação.

### **Observações:**

1. Critérios de avaliação:
  1. Corretude: 80%
  2. Interface: 10%
  3. Legibilidade: 10%
2. Todos os trabalhos devem possuir no cabeçalho a identificação dos autores.
3. Trabalhos copiados (com ou sem eventuais disfarces) terão a nota dividida pelo número de cópias (inclusive o original).
4. Trabalhos atrasados terão um desconto de 1 ponto por dia de atraso até o limite de 5 dias de atraso.
5. Trabalhos com erros de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
6. É muito importante que seu programa esteja bem comentado e bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa. A legibilidade do código será levada em consideração pelo critério de avaliação do trabalho.
7. Você deve enviar os arquivos fontes (.c e .h), além do *makefile*, em um arquivo compactado através da funcionalidade de atividades práticas do sistema acadêmico