



Trabalho Prático IV

Regras Básicas

1. extends Trabalho Prático 03
2. Fique atento ao Charset dos arquivos de entrada e saída.
3. em caso de empate das chaves, favor desempatar pelo campo `nome`.
4. para os campos do tipo `String` é recomendado usar o método `compareTo` para ordenação.



Steam é uma plataforma criada em 2003 pela Valve que revolucionou o mundo dos games. O serviço apresentou recursos inovadores como o download de jogos em formato digital, o desbloqueio de conquistas (achievements) e a interação entre usuários. Atualmente, milhões de pessoas utilizam o serviço todos os dias. O download do programa é de graça, mas muitos games são pagos, embora a Steam ocasionalmente libere alguns jogos grátis durante finais de semana. O serviço possui hoje aplicativos para Android e iOS e versões para Mac e Linux. Além disso, a versão do Brasil conta com preços em reais, e você pode pedir reembolso de jogos que não rodaram ou que não tenham atendido às suas expectativas.

Neste Trabalho Prático sua tarefa é organizar as informações dos jogos disponíveis para exibição ao usuário. Entretanto, esses dados estão espalhados em um arquivo `.csv` obtidos a partir de uma busca na plataforma. Todos esses dados estão agrupados no arquivo `games.csv`, e o mesmo deve ser colocado na pasta `/tmp/`.¹ Para isso, você deve ler, organizar e armazenar os dados de cada jogo em memória, utilizando as estruturas de dados em aula (Lista, Pilhas, Filas, Árvores e Hash). Em seguida executar as operações descritas nos arquivos de entrada. Muito cuidado ao realizar o *parser* do texto. Fique atento a descrição dos dados que serão lidos e manipulados pelo seu sistema.

¹Quando reiniciamos o Linux, ele normalmente apaga os arquivos existentes na pasta `/tmp/`.

Estruturas Flexíveis

1. **Lista com Alocação Flexível em Java:** Refazer a Questão “Lista com Alocação Sequencial” do Trabalho Prático II usando lista dinâmica simples.
2. **Pilha com Alocação Flexível em Java:** Refazer a Questão “Pilha com Alocação Sequencial” do Trabalho Prático II.
3. **Fila Circular com Alocação Flexível em Java:** Refazer a Questão “Fila Circular com Alocação Sequencial” do Trabalho Prático II. Lembre-se que essa fila terá tamanho máximo igual a cinco.
4. **Quicksort com LISTA DINÂMICA DUPLAMENTE ENCADEADA em Java:** Refazer a questão “Quicksort” do Trabalho Prático III com lista duplamente encadeada. O nome do arquivo de log será matrícula_quicksort2.txt.
5. **Pilha com Alocação Flexível em C:** Refaça a questão 2 deste TP na linguagem C.
6. **Quicksort com LISTA DINÂMICA DUPLAMENTE ENCADEADA em C:** Refaça a questão 4 deste TP na linguagem C.
7. **Matriz Dinâmica em Java:** Complete o código da classe matriz dinâmica visto na sala de aula. A primeira tarefa consiste em, no construtor da classe Matriz, dados os números de linha e coluna, fazer as devidas alocações de células. As demais tarefas são as implementações dos métodos Matriz soma(Matriz), Matriz multiplicacao(Matriz), void mostrarDiagonalPrincipal() e void mostrarDiagonalSecundaria(). A entrada padrão é composta por vários casos de teste sendo que o número de casos é um inteiro contido na primeira linha da entrada. Em seguida, temos cada um dos casos de teste. Cada caso é composto por duas matrizes. Para cada caso de teste, temos que suas duas primeiras linhas contêm um número inteiro cada representando os números de linhas e de colunas da primeira matriz, respectivamente. Em seguida, temos os elementos da primeira matriz que estão representados nas próximas l linhas onde l é o número de linhas dessa matriz. Cada uma dessas linhas têm c colunas onde c é o número de colunas dessa matriz. Nas duas linhas seguintes, temos os números de linhas e colunas da segunda matriz do caso de teste. As $l2$ linhas seguintes têm $c2$ colunas contendo os elementos da segunda matriz. $l2$ e $c2$ correspondem aos números de linhas e colunas da segunda matriz do caso de teste, respectivamente. A saída padrão contém várias linhas para cada caso de teste. As duas primeiras linhas de saída de um caso de teste correspondem às diagonais principal e secundária da primeira matriz, respectivamente. As demais ls linhas de um caso de teste correspondem as linhas matriz obtida pela soma das duas matrizes do caso de teste sendo que essas linhas contêm cs colunas referentes às colunas da matriz de soma. Da mesma forma, as linhas seguintes do caso teste contêm lm linhas com cm colunas representando os elementos da matriz de multiplicação onde lm e cm são os números de linhas e colunas da matriz de multiplicação.