

Algoritmos e Estruturas de Dados I

Professores: Isabel H. Manssour e Michael Móra

Trabalho 1 – Consultas sobre Acidentes de Trânsito em Porto Alegre

1 Descrição Geral do Trabalho

Este trabalho, que deverá ser feito **em dupla**, tem como objetivo modelar e implementar um sistema de consultas sobre os acidentes de trânsito de Porto Alegre usando estruturas encadeadas. Para isso, inicialmente, deve ser feita a leitura de um arquivo contendo informações sobre os acidentes, para posteriormente serem implementadas diferentes consultas sobre estes acidentes. Leia **atentamente** o enunciado do problema, identifique os elementos e estude as classes que deverão ser criadas para a implementação das consultas.

1.1 Estrutura de Dados e Formato do Arquivo

A primeira tarefa para a realização desse trabalho é criar uma estrutura de dados capaz de manipular as informações dos acidentes que forem lidas do arquivo. As informações dos acidentes de 2009 a 2012 estão em um arquivo CSV no Moodle que possui a seguinte estrutura:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	LOG1	TIPO_ACI	DATA_HO	DIA_SEM	FERIDOS	FATAIS	AUTO	TAXI	LOTACAO	ONIBUS_L	ONIBUS_I	CAMINHA	MOTO	CARROCA	BICICLETA	TEMPO	NOITE_DIA	REGIAO					
2	AV ASSIS	(ABALROAI	20090101	QUINTA-F	0	0	2	0	0	0	0	0	0	0	0	0	0 BOM	NOITE	NORTE				
3	R JOSE DE	ABALROAI	20090101	QUINTA-F	0	0	1	1	0	0	0	0	0	0	0	0	0 BOM	NOITE	SUL				
4	R EUSTAQU	ABALROAI	20090101	QUINTA-F	1	0	2	0	0	0	0	0	0	0	0	0	0 BOM	DIA	SUL				
5	AV FARRA	CHOQUE	20090101	QUINTA-F	0	0	2	0	0	0	0	0	0	0	0	0	0 BOM	DIA	NORTE				
6	AV PROTÁ	ABALROAI	20090101	QUINTA-F	0	0	2	0	0	0	0	0	0	0	0	0	0 NUBLADO	DIA	LESTE				
7	AV JOAO	(ATROPELA	20090101	QUINTA-F	1	0	1	0	0	0	0	0	0	0	0	0	0 BOM	NOITE	CENTRO				
8	AV PROF	(ABALROAI	20090101	QUINTA-F	1	0	2	0	0	0	0	0	0	0	0	0	0 BOM	NOITE	SUL				
9	AV VENA	(ABALROAI	20090101	QUINTA-F	0	0	1	1	0	0	0	0	0	0	0	0	0 NUBLADO	NOITE	CENTRO				
10	R GEN COI	(ABALROAI	20090102	SEXTA-FEI	0	0	2	0	0	0	0	0	0	0	0	0	0 NUBLADO	DIA	NORTE				

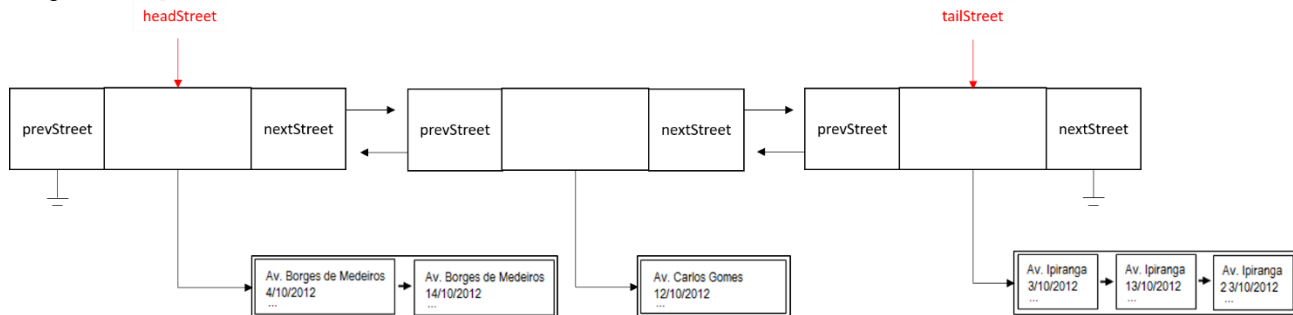
Este arquivo foi criado a partir das informações disponíveis em <https://dadosabertos.poa.br/dataset/acidentes-de-transito-vitimas>. Para este trabalho serão usados os dados de 2009 a 2012 com apenas alguns campos selecionados. Um exemplo de código para fazer a leitura deste arquivo também está disponível no Moodle.

Portanto, inicialmente analise estes dados para depois executar a primeira tarefa, que consiste na modelagem de uma estrutura encadeada para armazená-los. Esta estrutura consiste em uma lista que deverá ser mantida ordenada pelo nome da rua/avenida onde ocorreu o acidente. Também é preciso definir as classes e métodos que irão gerenciar esta lista, considerando as informações apresentadas a seguir.

Conforme ilustra a próxima figura, o nodo deverá ter referência para o próximo nodo e o nodo anterior que contém o acidente ordenado pelo nome da rua/av/trav. Além disso, cada nodo guardará a lista de acidentes que ocorreram na mesma rua/av/trav. Esta lista também deverá ter métodos que permitam navegar por esta estrutura e que garantam que cada vez que um novo acidente for armazenado, seja feito o encadeamento e o armazenamento correto para manter a ordenação por rua/av/trav.

prevStreet	Lista de Acidentes	nextStreet
------------	-----------------------	------------

A figura a seguir ilustra como os nodos devem ser conectados.



Depois disso, a partir da leitura do arquivo csv que foi fornecida, faça o correto armazenamento das informações lidas na estrutura encadeada implementada. É importante atentar para o fato de que os nomes das ruas/avenidas possuem “Av.” ou “Rua” na frente. Para facilitar o processo de ordenação, sugere-se que esta informação seja guardada de forma separada (por exemplo, um atributo guarda se é rua, avenida, travessa, etc., e o outro atributo guarda apenas o nome). Um exemplo de código para fazer esta separação é apresentado a seguir.

```
String s = "TRAV GERMANO GARCIA";
String sId = s.substring(0,s.indexOf(" "));
String sNome = s.substring(s.indexOf(" ")+1,s.length());
```

Para comparação de nomes use o método *compareTo*, como ilustra o exemplo abaixo:

```
String s1 = "Ipiranga";
String s2 = "Bento Gonçalves";
if(s1.compareTo(s2)<0)
    System.out.println(s1 + " vem antes de " + s2);
else if (s1.compareTo(s2)>0)
    System.out.println(s2 + " vem antes de " + s1);
else
    System.out.println(s2 + " eh igual a " + s1);
```

1.2 Consultas sobre a Estrutura

A partir da estrutura encadeada implementada, deve ser possível fazer as seguintes consultas através de uma interface textual simples:

1. Rua/av/trav na qual ocorreram mais acidentes;
2. Identificar para uma determinada rua/av/trav o dia da semana em que mais ocorreram acidentes;
3. Apresentar o total de acidentes envolvendo moto;
4. Permitir navegar pelos acidentes ordenados por rua/av/trav, isto é, avançar e retroceder, apresentando também o total de acidentes desta rua/av/trav.

1.3 Análise de Algoritmos

A análise de algoritmos também faz parte deste trabalho. Para isso, deverá ser respondido o questionário relacionado a este conteúdo que está disponível no Moodle junto da especificação do trabalho.

2 Entrega e Avaliação

Leia **com atenção** as instruções a seguir.

2.1 Entrega da Parte I

A primeira entrega consiste em:

- Responder individualmente o questionário sobre análise de algoritmos (cada aluno responde o questionário de forma separada e a nota para esta parte do trabalho será individual);
- Entregar um arquivo no formato PDF com o nome da dupla e a especificação de todas as classes, incluindo seus métodos e atributos, que serão implementadas.
- Deve ser preenchido o questionário e deve ser feito o *upload* do arquivo PDF através do *Moodle* **até a data e horário especificado para entrega da Parte I do trabalho.**

2.2 Entrega da Parte II

- Neste momento o trabalho deve estar todo implementado e cada dupla deverá **entregar apenas um arquivo zip contendo somente a implementação feita (somente os arquivos .java necessários para sua execução)**. O nome deste arquivo deve ser os nomes e sobrenomes dos alunos, da seguinte forma **nome1_ultimosobrenome1_nome2_ultimosobrenome2.zip**
- Deve ser feito o *upload* deste arquivo através do *Moodle* **até a data e horário especificado para entrega da Parte II do trabalho.**

2.3 Avaliação

- Serão averiguadas as respostas para o questionário e se a solução usa estruturas encadeadas, está completa, eficiente e correta, e a qualidade e clareza do código implementado.
- Os critérios de avaliação serão os seguintes:
 1. 1.5 ponto: Questionário (Parte I);
 2. 1.0 ponto: Especificação das classes com seus métodos e atributos (Parte I);
 3. 2.5 ponto: Correta implementação da estrutura encadeada com a ordenação por rua e a lista de acidentes da mesma rua (Parte II);
 4. 1.0 ponto: consulta da rua/av/trav na qual ocorreram mais acidentes (Parte II);
 5. 1.0 ponto: identificação do dia da semana em que mais ocorreram acidentes para uma determinada rua/av/trav (Parte II);
 6. 1.5 ponto: apresentação do total de acidentes envolvendo moto (Parte II);
 7. 1.5 ponto: navegação pelos acidentes ordenados por rua/av/trav, isto é, avançar e retroceder, apresentando também o total de acidentes desta rua/av/trav (Parte II).
- **Não serão aceitos trabalhos com erros de compilação. Programas que não compilarem corretamente não serão avaliados.**

2.4 Observações

- Não poderão ser usadas coleções do Java, tais como *ArrayList* e *LinkedList*. Deverão ser usadas obrigatoriamente as estruturas de dados estudadas em aula.
- Os trabalhos que NÃO FOREM ENTREGUES através do *Moodle* seguindo as regras estabelecidas, até o dia e horário especificado, receberão nota zero.
- Trabalhos que apresentarem CÓPIAS DAS SOLUÇÕES de outros colegas resultarão em NOTA ZERO para todos os alunos envolvidos.