

PROJETO SI200A – Algoritmos e Programação de Computadores II

Turmas A, Noturno

23 de setembro de 2016

1 Projeto

Em dupla. Data da entrega: veja no moodle.

Um método de identificação de adulteração no leite com uso de ultrassom recebe de um osciloscópio dois parâmetros de entrada: o coeficiente de atenuação de onda (*Atenua*) e a velocidade de propagação da onda (*Veloc*).

A adulteração assistida é feita pela adição de água em percentuais diferentes e pré-estabelecidos em cada amostra. Por exemplo, uma mesma amostra de leite é adulterada com 3%, 6%, 10%, 15% e 20% de água.

A hipótese é que as variáveis *Atenua* e *Veloc* sejam capazes de discriminar uma amostra de leite originalmente pura de outra amostra adulterada.

Originalmente, é executada uma série de medidas em laboratório. São criadas n amostras e cada uma das amostras sofre 5 adulterações com água, de 3% a 20% conforme já descrito. Cada amostra possui um *ID*.

Você receberá dois arquivos de dados, *Atenua.txt* e *Veloc.txt*, o primeiro contendo o valor da velocidade e o segundo o valor da atenuação.

Cada linha do arquivo refere-se a uma mesma amostra. Veja o exemplo abaixo:

Atenua.txt

Valor da atenuação por percentual de adição de água

ID;	3%;	6%;	10%;	15%;	20%
0;	-6.0439972;	-1.9241547;	-3.5895328;	10.7524626;	15.7773367 (amostra)
1;	3.2974902;	5.6128886;	0.8047239;	8.6452441;	18.0137429
2;	2.9151078;	-1.3317745;	-3.6975397;	10.1737065;	15.7718065
3;	6.0250198;	7.756944;	-5.3817805;	15.0544211;	14.1416601
4;	4.7763440;	0.7732832;	-3.6983976;	10.7874484;	15.5267889

Isso significa que a amostra com $ID = 0$ e adição de água de 3% possui *Atenua* = -6.0439972.

O segundo arquivo é semelhante, mas possui valores de velocidade da onda.

Veloc.txt

Valor da velocidade por percentual de adição de água

ID;	3%;	6%;	10%;	15%;	20%
0;	1576.76;	1576.76;	1575.46;	1572.85;	1570.25 (amostra)
1;	1587.30;	1572.85;	1575.46;	1571.55;	1571.55
2;	1578.07;	1576.76;	1574.15;	1572.85;	1570.25
3;	1578.07;	1588.63;	1574.15;	1572.85;	1567.66

4; 1578.07; 1575.46; 1575.46; 1572.85; 1570.25

A junção das duas tabelas nos diz que a amostra com $ID = 0$ e adição de água de 3% um par de medidas, $Atenua = -6.0439972$. e $Veloc = 1576.76$.

Desenvolva um sistema que receba os arquivos dois disponibilizados no formato .txt, leia as informações das variáveis e crie os registros.

1) Você deve criar os seguintes registros:

- Medida: Este registro contém dois doubles. Um chamado *Atenua* e outro chamado *Veloc*.
- Amostra: Este registro possui um vetor de medidas. Ele deve possuir um vetor `Medida * med[MAX1]` e uma variável que `top` que guarda o número de elementos no vetor. também deve possuir uma variável `int ID`.
- ConjAmostras: Este registro possui um vetor de *Amostra*. Ele deve possuir um vetor `Amostra * amo[MAX2]` e uma variável que `top` que guarda o número de elementos no vetor.

2) Você deve criar um construtor de registro para cada um dos registros:

- Medida: O construtor deve ter a seguinte assinatura `Medida * newMedida(float Atenua, float Veloc);`
- Amostra: O construtor deve ter a seguinte assinatura `Amostra * newAmostr(char * linha1, char * linha2);`. A linha passada como parâmetro vira dos arquivos *Atenua.txt* e *Veloc.txt*. Por exemplo, a amostra 0 receberá as seguintes linhas como parâmetro:

0; -6.0439972; -1.9241547; -3.5895328; 10.7524626; 15.7773367 e

0; 1576.76; 1576.76; 1575.46; 1572.85; 1570.25

A partir destas duas linhas, atribuir o $ID = 0$ e criar 5 medidas e colocar no vetor `med`, chamando a função `newMedida`.

- ConjAmostras: O construtor tem a seguinte assinatura `ConjAmostras * ConjAmostras(char * fileAtenua, char * fileVeloc);`. O construtor recebe como parâmetro os dois arquivo de dados. Faz a leitura de cada uma das linhas e, para cada linha cria uma amostra.

3) Você deve criar uma função que imprime cada registro.

- `void printMedida(Medida * medida)` : Imprime, com duas casas de precisão, a *Atenua* e a *Veloc*.
- `void * printAmostr(Amostra * a)`; Imprime todas as medidas de uma amostra em uma única linha. Deve chamar a função `printMedida`.
- `void ConjAmostras(ConjAmostras * set)`: Imprime uma amostra por linha. Deve chamar o método anterior.

3) calcule o desvio padrão e a média aritmética, para cada percentual de adulteração.