

TRABALHO PRÁTICO III

Objetivo

O objetivo deste trabalho é implementar um programa em java que explora a manipulação de arquivos e programação orientada a objeto.

Descrição

Este trabalho tem por objetivo aplicar os conceitos de Programação Orientada a Objetos, vistos em aula. O objetivo geral do programa a ser desenvolvido é gerenciar a matrícula de alunos em um conjunto de disciplinas. Todos os dados devem ser gravados em arquivos-texto e podem ser usados pelo programa diversas vezes.

Características do programa a ser implementado

O programa deverá executar, pelo menos estas funções:

- Escolher um aluno para trabalhar: o usuário deve informar o número de matrícula e a partir desta informação, todas as demais operações serão realizadas sobre este aluno;
- Matricular o aluno em uma disciplina: o usuário seleciona a disciplina em uma lista e a seguir a turma, também em uma lista. A partir disto sistema insere a mesma em seu horário, se não houver colisão de horário com outra disciplina. Caso haja colisão, a operação não deve ser realizada e uma mensagem de erro deve ser apresentada ao usuário;
- Retirar o aluno de uma disciplina: O sistema apresenta uma lista com as disciplinas em que o aluno está matriculado e o usuário escolhe qual deseja remover;
- Imprimir os alunos matriculados em uma disciplina: o usuário deve informar o código da disciplina e o sistema deve imprimir a listagem dos alunos (matricula e nome) em ordem alfabética do nome;
- Imprimir o horário do aluno: exibe na tela, o horário atualizado do aluno.

ARQUIVOS

Para poder funcionar o programa deverá ter acesso a alguns arquivos (no formato texto) com os dados de alunos e disciplinas existentes.

A seguir, cada uma destes arquivos é apresentado:

Arquivo de **DISCIPLINAS**: conterà, em cada linha o código, o número de créditos, o nome, a turma e o horário de cada uma turma de cada disciplina. Os dados de uma disciplina (que estão em uma linha) deverão ser separados por um asterisco '*', como no exemplo a seguir.

4115K*04*Álgebra Linear e Geometria Analítica
4613A*06*Algoritmos e Programação I
4613A*06*Algoritmos e Programação I

4613C*06*Algoritmos e Programação II
4613C*06*Algoritmos e Programação II
4622M*04*Avaliação de Desempenho de Sistemas
4115H*04*Cálculo A

Arquivo de **ALUNOS**: conterá, em cada linha a matrícula e o nome de cada aluno, como no exemplo:

052001344*ALEXANDRE KAZUO SEKI
051041374*ANDRE OURIQUE SOLARO
052001377*ANDREW SANTANA POMPEO
051041440*BERNARDO BOSAK DE REZENDE
052001385*BERNARDO DE LIMA BERRA

Um exemplo deste arquivo pode ser obtido neste link.

Arquivos de **TURMAS**: neste arquivo, cada linha deverá conter o código de uma disciplina, o número de créditos e o horário da turma, como no exemplo a seguir:

4613C*128*2*AB*4*CD*6*CD
4622M*138*2*FG*4*FG

Além dos arquivos de entrada, o programa deverá ser capaz de gerar dois arquivos:

Arquivo de **LISTA DE CHAMADA**: deverá existir para cada disciplina/turma um arquivo com todos os alunos matriculados nesta turma. O nome do arquivo é o código da disciplina. Nestes arquivos, cada linha deverá conter o nome e o número de matrícula dos alunos matriculados, como no exemplo do arquivo **4613G04.128**:

Disciplina: Algoritmos e Programação I
CODCRED: 4613C-04
Turma: 128

052001344 ALEXANDRE KAZUO SEKI
051041374 ANDRE OURIQUE SOLARO
052001377 ANDREW SANTANA POMPEO
051041440 BERNARDO BOSAK DE REZENDE

052001385 BERNARDO DE LIMA BERRA

Arquivos de **HORARIO** deverá existir um arquivo para cada aluno. O nome do arquivo é o número de matrícula do aluno e a extensão é **HOR**. Nestes arquivos, cada linha deverá conter o código de uma disciplina e a turma em que cada aluno está matriculado, como no exemplo do arquivo 051041374.HOR:

4115K*04*128

4613C*06*139

Considerações finais

O trabalho pode ser realizado individualmente. Este deve ser entregue até o dia determinado na agenda da disciplina, via moodle, antes do início da aula.

A entrega deve ser feita com um arquivo compactado (.zip) onde deverão estar contidos os arquivos do projeto bluej e um relatório (em PDF) descrevendo a solução implementada. O nome do arquivo deve conter nome e sobrenome do aluno. Ex.: AnaMattos.zip.

O material postado no moodle é de inteira responsabilidade do aluno. A presença de arquivos corrompidos, que impeçam a avaliação do trabalho pelo professor será considerada como a não entrega do trabalho.

Bom trabalho