/\*\*  
\* UFRJ - Universidade Federal do Rio de Janeiro  
\* IM - Instituto de Matemática  
\* DMA - Departamento de Matemática Aplicada  
\*  
\* TMAB - Programação C/C++ & Banco de Dados  
\* ParteII - Programação OO em C++  
\*   
\* Descrição do Estudo Dirigido 04  
\* Prazo de Entrega: 21 de novembro de 2017  
\*  
\* Prof. Milton Ramirez (milton@labma.ufrj.br)  
\* Rio de Janeiro, 13 de novembro de 2017  
\*\*/

Desenvolver um programa que gere arquivos CSV para popular a base de dados desenvolvida no ED2.   
O programa gera dados baseado em lista de entrada (lidos de outros arquivos) e de dados gerados de forma randômica. A saída do programa é uma sequencia de arquivo no formato CSV que serão usados para popular a base de dados modelada no ED2.  
O arquivo CSV é um arquivo texto, onde cada linha contém os dados de cada campo de uma ou mais tabela, separados por vírgula ou ponto e vírgula.

Por exemplo, para a tabela Pessoa, criada pelo script.sql

CREATE TABLE Pessoa  
(  
NOME CHAR(256),  
IDADE NUMBER,  
SEXO CHAR(1)  
)

O arquivo CSV correspondente seria  
João da Silva; 22; M  
Maria das Dores; 32; F  
...

A partir da especificação do ED02, o programa deve gerar arquivos CSV para preencher tabelas relacionadas a:  
1. Cursos regulares de graduação (Engenharias, Direito, Matemática, Medicina, Educação Física, Letras e Artes etc.)  
2. Período letivo (dois semestres por anos)   
3. Disciplinas oferecidas por grade e em cada semestre/ano.   
4. Turmas oferecidas por semestre/ano  
5. Grade curricular por curso  
6. Horários disponíveis (por sala?)  
7. Sala, laboratório, gabinete, prédios etc.  
8. Alunos regularmente matriculados   
9. Professor (de turma, coordenador, orientador acadêmico, orientador IC)  
10. Inscrição de um aluno por turma com nota final  
11. Atividades de extensão ou de iniciação científica.   
12. Trabalho apresentado na JIC  
13. Jornadas de Iniciação e Extensão.  
Alguns dessas tabelas podem ser preenchidas pelo mesmo arquivo, ou seja, não precisa fazer um arquivo para cada um dos item citados.   
No caso de lista de nomes, esses podem ser gerados combinando nomes lidos de uma lista (arquivo) de entrada que contem três (ou mais) colunas relativas ao Primeiro Nome, Nome do Meio, Sobrenome.  
Exemplo de Arquivo de entrada:  
1) Maria, Silva, Albuquerque  
2) Ana, Marcondes, das Dores  
Exemplo do arquivo de saída CSV (para o CREATE TABLE anterior):  
Maria Silva Albuquerque; 32; F  
Maria Silva das Dores; 21; F  
Maria Marcondes Albuquerque; 22; F  
Maria Marcondes das Dores; 56; F  
Etc...

Outros arquivos podem ser gerados do mesmo jeito e até usando uma função randômica para combinar números a fim de gerar com frequências desiguais, por exemplo, salas por prédio (número e capacidade). Pode-se colocar os valores em um vector e depois sortear entre as turmas.

/\*\*  
\* UFRJ - Universidade Federal do Rio de Janeiro  
\* IM - Instituto de Matemática  
\* DMA - Departamento de Matemática Aplicada  
\*  
\* TMAB - Programação C/C++ & Banco de Dados  
\* ParteII - Programação OO em C++  
\*   
\* Descrição do Estudo Dirigido 05  
\* Prazo de Entrega: 30 de novembro de 2017  
\*  
\* Prof. Milton Ramirez (milton@labma.ufrj.br)  
\* Rio de Janeiro, 17 de novembro de 2017  
\*\*/

ED05 – Carga da Base de Dados e Consultas Gerais  
Prazo de entrega: 30 novembro de 2017  
Pretende-se carregar o banco de dados modelada e criada no Estudo Dirigido 02 a partir da sequencia de arquivos CSV gerados no Estudo Dirigido 04 e realizar uma série de consultas em SQL.  
O resultado a ser entregue é composto dos “scripts de carga” e a lista de consultas em SQL, sendo que para algumas das consultas também deverá ser entregue um arquivo TXT com o resultado da consulta.  
Usando o programa desenvolvido no ED04 e editando na mão alguns dos arquivos CSV, gere uma base de dados para carga com as seguintes restrições:  
a) Tenha pelo menos três cursos de graduação cadastrados, cada qual com no mínimo 02 anos de duração (04 semestres).  
b) Tenha pelo menos 20 disciplinas básicas (comum a todos os cursos), 12 profissionais e 8 optativas;  
c) Dois anos com dois semestres cada;  
d) Dentre as disciplinas básicas deve haver algumas com mais de duas turmas no mesmo semestre/ano;  
e) Tenha professores o suficiente para que cada um não lecione mais do que quatro disciplina por semestre  
f) Tenha alunos distintos o suficiente para que as turmas tenham pelo menos dez alunos inscritos por semestre dos cursos  
A lista de consultas é a seguinte:  
Obs: onde se lê “de um(a) dado(a) X”, substitua por algum valor de dado existente em seu banco de dados.  
1) Quais os cursos de graduação e suas cargas horárias (número de semestres)?  
2) Quais as disciplinas de 4 créditos de um dado curso ?  
3) Pauta: Quais alunos estão inscritos em uma dada turma?  
4) Quais as disciplina de um dado período de um dado curso?  
5) Quais as disciplinas (turmas) oferecidas em um dado ano/semestre?  
6) Quais as disciplinas (turmas) oferecidas em um dado ano/semestre, relativas a um dado período do curso de um dado aluno?  
7) Qual a sala, horário e professor de uma dada turma?  
8) Quantos alunos estão inscritos em cada turma de um dado ano/semestre?  
9) Quantos alunos estão inscritos em cada turma de um dado professor em um dado ano/semestre?  
10) Quem é o aluno mais velho ?  
11) Qual a distribuição etária, por sexo, dos alunos?  
12) Qual a grade curricular por período de um dado curso?  
13) Qual o histórico de um dado aluno?  
14) Dado dois semestre/anos distintos, quais disciplinas foram oferecidas em ambos?   
15) Dado dois semestre/anos distintos, quais disciplinas foram oferecidas em um deles e não no outro?  
16) Quantos alunos têm em comum dois dados professores em um dado semestre/ano?  
17) Quais as médias por turma ou disciplinas, por semestre/ano?   
Para cada uma das seguintes consultas deve ser entregue os seus resultados:  
1. Totais de alunos, de professores e de disciplinas  
2. Total de turmas por semestre/ano  
3. Total de alunos inscritos em turmas por semestre/ano  
4. Totais de disciplinas por curso

Post atualizado sobre como usar a OTL4.0

Dicas para o ED6 - Construção de Aplicativo de Banco de Dados para Inscrição de "Alunos" em "Turmas"  
Para os que estão com problemas em fazer o Driver ODBC do MySQl funcionar, segue as dicas:   
Desenvolvendo uma aplicação de Banco de Dados em C++, usando CodeBlocks, Template Library C++/SQL OTL e conexão ODBC com o MySQL.  
Essa documentação foi escrita usando a versão 5.7.14 de 64 bits do MySQL, instalado via WAMPServer 3.0.6., com “MySQL ODBC Connector” versão 5.3.8 de 32 bits, instalada via msi, e Template Library C++/SQL OTL versão 4 (4.0.404). Tudo compilado no CodeBlocks 16.1 via MinGW de 32 bits, versão 4.8.   
O MySQL de 64 bits roda tanto com o drive ODBC de 32bits como o de 64bits.Porém o programa C++ tem que ser ligado (link edition) com o conector ODBC correspondente, senão ao rodar irá apresentar erro de “incompatibilidade de arquitetura entre o ODBC e a aplicação”. Assim, como nesse exemplo apresentado é utilizado o compilador MinGW de 32 bits, o conector ODBC utilizado foi o de 32 bits também.   
Para usar o conector ODBC de 64 bits (“MySQL ODBC Connector” = mysql-connector-odbc-5.3.8-winx64) basta trocar a versão do compilador MinGW, que o CodeBlocks usa, para a sua versão de 64 bits. Esse compilador pode ser encontrado em: <https://sourceforge.net/>…/x64-4.8.1-release-posix-…/download   
1) Os sites a partir dos quais foram baixados esses produtos foram:  
a) WAMP Server: [https://sourceforge.net/projects/wampserver/](https://l.facebook.com/l.php?u=https%3A%2F%2Fsourceforge.net%2Fprojects%2Fwampserver%2F&h=ATOKU7UfmvxtT8glIVHlVfOBZjgl5BLV-jPo3k7NcX5VBPkPJizKFQky17BcWMzVWq_oayMWNyrh1cZ1m9RoeeIMyrY8YPsmDBJopPd4qTl4msBRSsax2olAb1CgoE6HAkcssy9nZD0Ba1TsxIeYmy7iPSGBCcum8r_dZS73LZlVF3N2tJ1nGtfvFBuw-XdNi3RrfK6ddl7eQFhrKE3GV6LlhJFpZKf0uoza4r23AFm2tt4MzkSxieALqALVBdpwo8yliOp72n1v4ggK_jfkZfvYw0AyeM6WP3k)  
b) MySQL ODBC Connector:<https://dev.mysql.com/downloads/connector/odbc/>. Para esse é necessário criar, ou ter, uma conta no Oracle Developer.  
c) CodeBlocks: <http://www.codeblocks.org/downloads/>, versão com compilador  
d) TL C++/SQL OTL: <http://otl.sourceforge.net/home.htm>  
2) Configuração  
Após baixar e instalar todos os produtos acima listados, devemos configurar as seguintes coisas:  
a) Copiar o head da OTL do zip para o diretório de include do compilador C++ sendo utilizado. No exemplo aqui listado é o MinGW no CodeBlocks. Assim, devemos copiar o arquivo “otlv4.h” do zip da OTL baixado de<http://otl.sourceforge.net/otl3_down.htm> para o diretório “C:\Program Files (x86)\CodeBlocks\MinGW\include”, isso supondo que o CodeBlocks foi instalado pela opção default. Senão, com mencionado, deve-se procurar o diretório de include o compilador C++, ou indicar ao compilador onde achar esse head da OTL.  
b) Configurar o CodeBlock para usar o ODBC:  
i) Configurar o CodeBlocks para achar a biblioteca do conector ODBC. Com o botão direito do mouse sobre o nome do projeto que se deseja usar ODBC, na aba “Projects” do “Manager” (shift-F2, se não estiver na tela), selecionar a opção “Build Options...” que abrirá a janela “Project build options”. Na aba “Linker setting”, no painel “Link libraries” escolher o botão “Add” e adicionar um caminho até a biblioteca “myodbc5a.lib” (que na instalação default fica no diretório dos programas de 32 bits “C:\Program Files (x86)\MySQL\Connector ODBC 5.3\”).   
ii) Ligar a diretiva para o Linker usar o biblioteca ODBC: na mesma aba do item anterior, no painel “Other linker options” digitar “-lodbc32”.  
iii) Configurar o diretório da biblioteca dinâmica “myodbc5a.dll”: Na mesma janela de “Projects build options” do “Manager”, na aba “Search directories”, na sub aba “Linker”, botão “Add”, coloque o diretório citado no item “(i)” acima, ou seja, o diretório onde o MyODBC Connectorde 32 bits foi instalado, que na instalação default é “C:\Program Files (x86)\MySQL\Connector ODBC 5.3\”).  
c) Configurar uma nova fonte de dados ODBC, ou seja o DNS:   
i) Acionar o “Configurador de fonte de dados ODBC” correspondente a versão que foi instalada (32 ou 64 bits). Esse configurador pode ser achado na janela de “Ferramentas Administrativas” do Windows que, em geral, é acessada via “Painel de Controle”, ou pode ser acessada pela busca no Windows com a string “ODBC”. Os sistemas Windows corretamente instalados irão apresentar as duas versões “Fonte de Dados ODBC (32 bits)” e “Fonte de Dados ODBC (64 bits)”. Nesse exemplo usamos a primeira.  
ii) Ao acionar o “Administrador de Fontes de Dados ODBC”, vá para a aba “DNS de Usuário” e acione o botão “Adicionar”. Escolha o drive “MySQL ODBC 5.3 ANSI driver” (se for usar caracteres Unicode, uso o correspondente).  
iii) Configurar a fonte: Surgirá a tela “MySQL Connector/ODBC Data Source Configuration”. Nela coloque um nome para a sua fonte de dados no campo “Data Source Name”. Esse nome será usado no programa. No campo TCP/IP coloque “localhost” (ou o endereço IP do MySQL se estiver em outra máquina). A porta padrão é a “3306”. O campo user e password deve ser o mesmo do banco MySQL que você instalou (em geral é root/root, ou root sem senha). Na lista do campo “Database”, escolha o banco onde o programa irá realizar suas operações (se ainda não existir, pode-se criar usando o comando “CREATE DATABASE <nome do banco>” na linha de prompt do console MySQL ou na página do PHPAdmin). Teste a conexão antes de sair para ver se tudo está correto.  
iv) Qualquer dúvida, há várias páginas na web sobre esse assunto.  
3) Compilando e rodando  
Para verificar se tudo está correto, copie o exemplo da página do OTL relativo a conexão genérica via ODBC ((OTL 4.0, Exemples for Generic ODBC. Exemple 21 (Simple Insert/Select) que pode ser encontrado em<http://otl.sourceforge.net/otl3_ex21.htm>.  
Lembre-se de editar as informações de conexão para utilizar o DNS configurado no passo (2.c.iii) anterior. Em geral está na linha 64, comando “db.rlogon("UID=root;PWD=;DSN=nome do DNS configurado)”;  
A partir desse exemplo pode-se desenvolver o ED06.

/\*\*  
\* UFRJ - Universidade Federal do Rio de Janeiro  
\* IM - Instituto de Matemática  
\* DMA - Departamento de Matemática Aplicada  
\*  
\* TMAB - Programação C/C++ & Banco de Dados  
\* ParteII - Programação OO em C++  
\*   
\* Descrição do Estudo Dirigido 06  
\* Prazo de Entrega: 07 dezembro de 2017  
\*  
\* Prof. Milton Ramirez (milton@labma.ufrj.br)  
\* Rio de Janeiro, 17 de novembro de 2017  
\*\*/

ED06 – Sistema de Informações SID  
SID – Sistema de Inscrição em Disciplina  
Desenvolver um sistema em C++, usando a “Template Library OTL ([http://otl.sourceforge.net/](https://l.facebook.com/l.php?u=http%3A%2F%2Fotl.sourceforge.net%2F&h=ATP3oC3hTb9zOJQeEkFQiPTZW88lw8VI-OTqLfODwwbKYuiy6dqVFBPbTimQA3iGTsAxtN55O-qn580JSUzszEhmUNpHuXsHmegGdUfR9K_Pvk-N_zEIcfZaUSm6OzMkH1WhygQBLtlrGtpQqSWWeXaHKJvk2x27hv12Vsz7x2y2nO60TJbckXRaNRpyoqX9DjCOG9SwiyNpGHWV-Lo3DxUAJv8R5GOzF8ILzgRxh9cB1PAcLcGT10QqzdLnyEQy18gJ-gpuvC88Sv6MgJjLzYO70Ws4b1-a-dwKtAAH) home.htm)”, que permita alunos devidamente matriculados realizar suas inscrições nas turmas das disciplinas ofertadas em um dado semestre. Também o sistema deve permitir a geração de certos tipos de relatórios. O resultado a ser entregue é, como usual, o código fonte em C++.  
O sistema deve permitir:  
1. A um aluno, identificado pelo DRE ou nome, escolher dentre as turmas das disciplinas oferecidas em um dado semestre/ano, uma na qual deseja se inscrever. As turmas ofertadas no semestre são identificadas pelo nome da sua disciplina e horário. O sistema deve informar sala, ementa e professor da turma/disciplina. Uma extensão seria permitir também a seleção de turmas pelo seu numero, professor ou sala. Ao final o sistema deve gravar na base a inscrição do aluno na turma escolhida.  
2. A um professor listar a pauta de uma de suas turmas em um dado semestre/ano.   
3. A um professor lançar as notas de uma turma.  
4. A um aluno, listar o seu histórico.   
Para os mais experientes pode-se melhorar a interface do item (1.) listando apenas as disciplinas que o aluno ainda não cursou e para as quais já possui o pré-requisito, indicando aquelas que estão oferecidas no mesmo horário. Pode-se permitir filtrar pelo seu curso ou não (assim verá as disciplinas de outros cursos). Para os itens (2.) e (3.) uma lista para que o professor escolha a turma dentre as que ele está ministrando no dado semestre/ano, a fim de evitar que ele tenha que se lembrar do código das suas turmas.