CP465, Database II Term Project Object Query Language, OQL

Instructor: I. Kotsireas e-mail ikotsire@wlu.ca
Project due date: **December 4, 2016**
Students may form teams (of no more than 4 students each) to work on the project collectively.

Introduction

The objective of this project is to build an implementation of the Object Query Language, OQL, (as specified by the ODMG, `http://www.odmg.org/`) on top of a Relational Database Management System, RDBMS. In particular, the user should be able to type an OQL query pertaining to a hypothetical Object Database and the program should:

1. `analyze` the query lexically
2. `parse` the query
3. `translate` the query into an "equivalent" SQL query
4. `execute` the SQL query on the underlying Relational Database

Object Query Language, OQL

The syntax of OQL will be given to us in the form of a BNF grammar. The OQL Grammar used in this project is taken from the book **The Object Data Standard : ODMG 3.0**, edited R. G. G. Cattell, Douglas K. Barry, Mark Berler, Jeff Eastman, David Jordan, Craig Russell, Olaf Schadow, Torsten Stanienda, and Fernando Velez, chapter 4, paragraph 12. The BNF grammar for OQL will be specified in an additional document. The four phases described above are detailed one by one below.

1 The lexical analysis phase, Lex

Identify the tokens in the BNF grammar of OQL and write a Lex program to recognize them. Keep in mind that the ulterior aim is for this program to cooperate with the Yacc parser.

2 The syntactic analysis (parsing) phase , Yacc

Transform the OQL BNF grammar into a Yacc program to produce the parser. It is a good idea to implement first a smaller subset of the grammar, compile the program, run it with several examples to make sure that it works properly, then continue with implementing some more production rules etc. If you choose to implement the grammar all at once, you will most probably have to go through a painful debugging phase.

3 The translation phase

Once the OQL query has been lexically analyzed and parsed, it will be translated to an SQL query. Details on how to perform this translation will be specified in an additional document.

4 The execution phase, JDBC/ODBC

The SQL query obtained from the translation phase, can then be executed on the underlying Relational Database using for example JDBC or ODBC.

Note: Lex and Yacc are two programs that work together to perform lexical and syntactic analysis. These are the programs that were presented in class. You are not obliged to use these tools for the project. You may use any other tools for lexical and syntactic analysis that you are familiar with, provided that they offer suitable interfaces that will allow you to complete the subsequent translation and execution phases of the project.

Submission of your work

By the project due date, each group is invited to submit the following items:

1. The `cover sheet`, (Names/Student IDs of group members, Course Number, Date)
2. The `design document`, which should include:
   (a) description of what features work and what features don't.
   (b) description and justification of the design choices made.
   (c) description of data structures used.
   (d) instructions on how to compile and run the code.
3. The `test document` which should demonstrate the correctness of the program through the successful treatment of some benchmark OQL queries that will be specified in an additional document.
4. The `code` in printed version as well as in electronic form. Well-commented code will count for more marks than code without comments.