Concordia University
Department of Computer Science
and Software Engineering

Comparative Study of Programming Languages
COMP 6411 --- Winter 2013

Term paper assignment

Deadline:	Wednesday April 10th, 2013
Evaluation:	paper (20%), presentation in class (10%)
Late submission:	not accepted

Problem statement

Compare and contrast various properties, features, paradigms and execution schemes and strategies used by a variety of programming languages and their suitability for particular tasks.

Groups

The project is to be done in groups of exactly 5. Each team will be provided with a group account to coordinate their group activities and manage the project's content (e.g. set up a CVS repository) to share documents, code and resources related to the group's project.

List of languages proposed (mandatory in bold)

C
ML
C#
Python
Smalltalk

Java
C++
Pascal
AspectJ
Ada

Scheme
PHP
Haskell
Objective-C
VisualBasic

Fortran
JavaScript
Perl
Ruby
Prolog

Proposal

By Wednesday February 13th, a short proposal must be submitted to the EAS under the label “proposal”. The proposal must describe:

- The list of the five members of the teams, including full name, student ID, and ENCS account ID.
- The list of languages chosen, specifying what compiler/IDE is to be used to do the programming parts of the term paper requirements. Each member of the team must be allocated a pair of two languages for which s/he will be responsible on behalf of the team.
- A preliminary list of references to be used for the comparative study between the chosen languages. At least 10 references must be given for each language chosen. The quality of the references will be graded.

Comparison Criteria

Language description Describe each language chosen: (1) the history of its development, including important contributors, dates, and important developments; (2) what programming paradigms are used by the language and its different versions; (3) explain how each language chosen is compiled and executed, including the typing strategy used, strategies used for memory management, etc; (4) highlight the strengths and weaknesses of the language; (4) highlight specific application domains for which this programming language is used. Explain for what reasons it is used for this particular application domain. (see individual assignment #4)

Availability of compilers/IDEs List all known compilers or integrated development environment currently available for the language. List on which operating system version these are available. These may include commercial, free or open-source solutions. Give the web address of each compiler/IDE available to download from the internet. Describe the compiler/IDE used for the programming part of the term project.

Runtime Efficiency Find sorting algorithms that can be implemented in all the languages chosen. Implement a program for each of these algorithms in each language chosen and measure the time taken and memory consumed for the execution of these programs. Compare the results and explain the differences. (see individual programming assignment #1)

GUI development List and briefly describe all known solutions to develop graphical user interfaces in the development of applications programmed using each chosen language. These may be either integrated in an IDE, or using commercial or open-source libraries. Choose five of your selected languages and implement a GUI component that allows the user to use the sorting algorithms and observe their functioning. (see individual programming assignment #2)

Database connectivity For each chosen language, list and briefly describe all known solutions to develop programs using an external database. Choose five of your selected languages and implement a program that read/writes data from/to an external database. Measure the read/write times for each solution using varied data sets. Discuss your results. (see individual programming assignment #3)

Structure of the Paper

Cover page Start with a cover page that states the composition of the team (full names and student IDs), the course name and number, and the list of programming languages being compared.

Table of Contents Following the cover page should be a table of contents that includes all the sections described below. All pages should be numbered and pages numbers be referenced in the table of contents.

Part 1: Description of each language There should be one separate section for the description of each language chosen. Each language should be described using the criteria described above. In each language's description section, there should be one subsection for each criteria.

Part 2: Overall Comparison Include graphs and tables whose aim is to compare all the languages according to the different criteria presented above. Examples for each criteria include and are not limited to:

History: Integrated timeline graph depicting the compared lifetime of the chosen languages.

Language features: Set of tables comparing all the languages according to different features including the typing strategy used, memory management strategy, programming paradigms, execution model used, etc.

Runtime efficiency: For each sorting algorithm used, a graph integrating all the run-time results for all the languages, both in terms of execution times and memory management.

Database throughput: Graph integrating all the run-time results for the five languages selected for database throughput comparison.

References Include an integrated list of references. Each reference should be numbered, and be referred to in the text using this numbering scheme. References should be complete, including title, authors names, publisher name, and year of publication. Quality of references is of prime importance. The highest valuable references are peer-reviewed publications: scientific journal papers, then conference proceedings papers, then books. Other non-peer reviewed sources have lower value. Non-peer-reviewed sources that do not themselves have references are considered to have zero value.

Referencing

All term papers must be strongly backed-up by relevant references used throughout the text. Failure to provide good references will result in major grade deductions. Plagiarism will be thoroughly checked and any plagiarism will result in major grade deductions and possibly a failure grade.

Presentations

A schedule is available on the course web site for the students to schedule their presentation time. Available dates for the presentations are March 27st, April 3rd, and April 10th 2013. During the presentations, you have to show your overall results of the comparison of the 10 chosen languages, demonstrate the usage of all programs developed, and present and explain the programming involved as well as the execution results.

Submission

No later than midnight on Wednesday April 10th 2013, the final term paper must be submitted to the EAS under the label “project 1”. A zip file containing all the developed code must be submitted to the EAS under the label “project 2”.