
**COMP 6411**

**Comparative Study of Programming Languages**

**Summer 2013**

# POTUM: THE BEVERAGE COASTER PROJECT

**Deliverable 3**

**Team D**

List of Team Members:
**Aravindan Balasubramanian**
**Lin Zhu**
**Romil Bandi**
**Safan Maredia**
**Shivangi Rathore**
**Qing Sun**

# Table of Contents

## 1.0 ABSTRACT

This simple report is written for the comparison of different POTUM programs. It firstly gives a general introduction of the programming languages comparison and the language evaluation criteria. Then it show the result of the comparison by a table which is based on some criteria for evaluating programming languages.

## 2.0 INTRODUCTION

Programming languages are used for controlling the behavior of a machine (often a computer). Like there are thousands of programming languages and new ones are created every year. Few languages ever become sufficiently popular that they are used by more than a few people, but professional programmers may use dozens of languages in a career.

So it is very meaningful and useful for us to compare with different languages. In this case, we use 6 different languages for solving one common problem. Besides, all 6 programs are based on the same algorithm. So the performances for different languages can easily been seen. But people can never say which language is good or bad since they may perform totally different in different cases.

## 3.0 PROGRAMS COMPARISON

Here is a table of the comparison of different POTUM programs.

The numbers from 1 to 6 in the blanks is the ranking number among the 6 programs. For example, Matlab and Python both get 6 since both of the two programs have better simplicity. All the details of the comparisons will be listed after the table.

| | **C++** | **JavaScript** | **Matlab** | **PHP** | **Python** | **Ruby** |
|---|---|---|---|---|---|---|
| **Simplicity** | 4 | 2 | 6 | 4 | 6 | 1 |
| **Orthogonality** | 6 | 5 | 1 | 2 | 3 | 4 |
| **Data Types** | 6 | 5 | 2 | 4 | 4 | 2 |
| **Syntax Design** | 5 | 3 | 3 | 3 | 5 | 6 |
| **Support for Abstraction** | 3 | 3 | 5 | 5 | 3 | 6 |
| **Expressivity** | 4 | 5 | 6 | 4 | 1 | 2 |
| **Type Checking** | 2 | 2 | 5 | 3 | 4 | 6 |
| **Exception Handling** | 6 | 5 | 1 | 4 | 3 | 2 |
| **Restricted Aliasing** | 2 | 6 | 4 | 3 | 5 | 3 |
| | | | | | | |
| **Total Mark** | 38 | 36 | 33 | 32 | 34 | 32 |

**Simplicity:**

The overall simplicity of a programming language strongly affects its readability. A language with a large number of basic constructs is more difficult to learn than one with a smaller number.

For certain programs, it is better to have less lines and basic constructs when they perform a common problem with a same algorithm.

Here are the numbers of lines for different codes of the different programs.

Matlab 87, Python 92, C++ 115, PHP 117, JavaScript 167, Ruby 209

The code of Matlab and Python are much shorter than others. And their performances are better in simplicity, following by C++ and PHP. Then it is JavaScript and Ruby.

**Orthogonality:**

Orthogonality is an important concept, addressing how a relatively small number of components can be combined in a relatively small number of ways to get the desired results. It is associated with simplicity; the more orthogonal the design, the fewer exceptions. This makes it easier to learn, read and write programs in a programming language.

For certain programs comparison, what we focus on is the number of components can be combined within a program. Under this standard, here are the performances for 6 programs.

The program with C++ performances perfect and the one using JavaScript is also very nice, following by a good program with Ruby. Remaining ones are with Python, PHP and Matlab.

**Data Types**

The presence of adequate facilities for defining data types and data structures in a language is another significant aid to readability.

The program with C++ and JavaScript do perfect in defining data types .The ones with PHP and Python are also very good. Ruby and Matlab are well.

**Syntax Design:**

The syntax, or form, of the elements of a language has a significant effect on the readability of programs.

Ruby has succinct and flexible syntax that minimizes syntactic noise and serves as a foundation. Python and C++ supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. The programs with them have well syntax design.

**Support for Abstraction:**

Briefly, abstraction means the ability to define and then use complicated structures or operations in ways that allow many of the details to be ignored. Abstraction is a key concept in contemporary programming language design.

Ruby is a thoroughly object- oriented with inheritance mixins and Meta Classes. PHP 5 also introduces abstract classes and methods. Both of them include the use of abstraction.

**Expressivity:**

Expressivity in a language can refer to several different characteristics. More commonly, it means that a language has relatively convenient, rather than cumbersome, ways of specifying computations.

Matlab is especially designed for matrix computations: solving systems of linear equations, computing eigenvalues and eigenvectors, factoring matrices, and so forth. For JavaScript, intuitively and easily readable constructs, complex expressions are compactly expressed. Their programs have different ways of specifying computations.

**Type Checking:**

Type checking is simply testing for type errors in a given program, either by the compiler or during program execution. Type checking is an important factor in language reliability. Because run-time type checking is expensive, compile-time type checking is more desirable.

Ruby supports dynamic typing and duck typing. And Matlab will also provide type checking before the execution. Python uses duck typing and has typed objects but untyped variable names. Their programs have a better performance in this aspect.

**Exception Handling:**

The ability of a program to intercept run-time errors (as well as other unusual conditions detectable by the program), take corrective measures, and then continue is an obvious aid to reliability. This language facility is called exception handling.

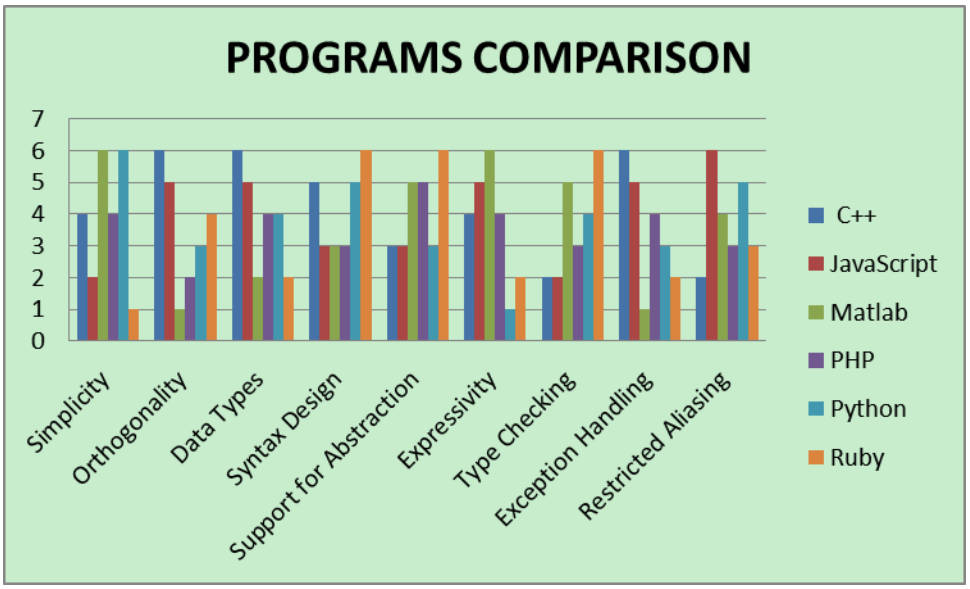
C++ and JavaScript have extensive capabilities for exception handling. Their programs are also provided good exception handling ability which can be seen from the case testing result. Both of them are the best performance ones.

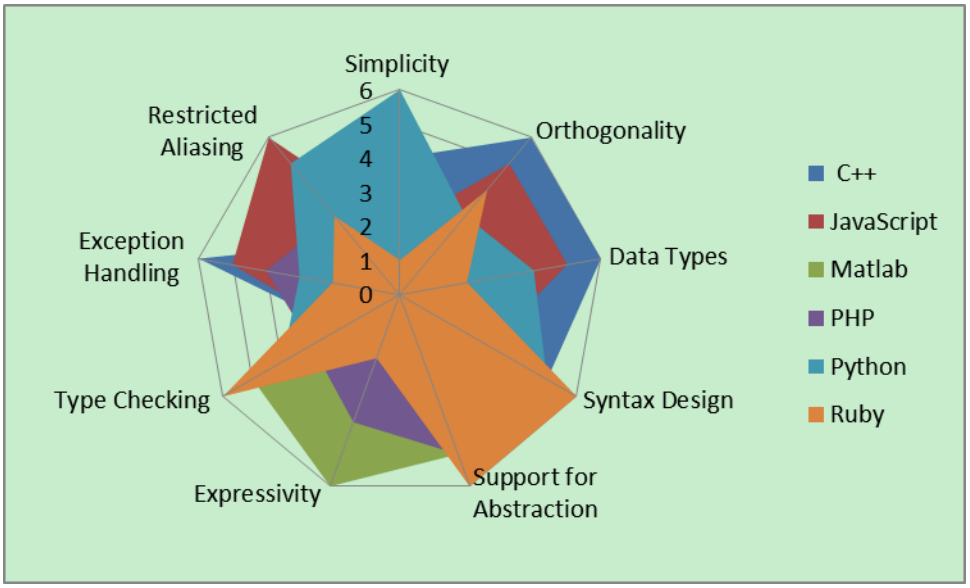
**Restricted Aliasing:**

Loosely defined, aliasing is having two or more distinct names that can be used to access the same memory cell.

Aliasing is supported in JavaScript and be restricted in the program of it.Restricted aliasing is provided well in Matlab. In its program, aliasing will always be used very carefully and seriously.

Here are two graphs which show the comparison results in a vivid way. It shows all 6 programs with different languages have their superiorities and weakness as well.




## 4.0 CONCLUSION

With respect to POTUM COMPARISON for Deliverable 3, comparison of different POTUM programs were carried out based on language evaluation criteria(Sebesta,2012,Section 1.3) Readability, Writability, Reliability, Cost described with characteristics. Comparison table listed above is expressive and conforms to our comparison standard used.

## 5.0 REFERENCES


En.wikipedia.org. 2005. *Comparison of programming languages - Wikipedia, the free encyclopedia*. [online] Available at: http://en.wikipedia.org/wiki/Comparison_of_programming_languages [Accessed: 31 Jul 2013].

Rigaux.org. n.d.. *syntax across languages (One Big Page)*. [online] Available at: http://rigaux.org/language-study/syntax-across-languages.html [Accessed: 31 Jul 2013].

Scriptol.com. 2013. *Popular programming languages*. [online] Available at: http://www.scriptol.com/programming/choose.php [Accessed: 31 Jul 2013].

wikiHow. n.d.. *How to Write a Language Analysis*. [online] Available at: http://www.wikihow.com/Write-a-Language-Analysis [Accessed: 31 Jul 2013].

Bolton, D. (1950). *Comparing Popular Programming Languages*. [online] Retrieved from: http://cplus.about.com/od/introductiontoprogramming/a/comparelangs.htm [Accessed: 31 Jul 2013].

Jvoegele.com (n.d.). *Programming Language Comparison*. [online] Retrieved from: http://www.jvoegele.com/software/langcomp.html [Accessed: 31 Jul 2013].