Assignment 1

CS 350

Due: September 21st, 2018

Jennifer Herasymuik

SID: 200352062

Chapter 1

1. (30 marks). Write an evaluation of a language you know, using the criteria listed below.

Criteria for Language Design Evaluation

a) Efficiency (translation and execution)

b) Simplicity (readability and writability)

c) Orthogonality

d) Syntax design

e) Reliability (type checking and exception handling)

f) Abstraction facilities (data and procedural)

1. Briefly define each criterion

2. Assign a letter grade (A through E) for the language for each of the criterion point and provide written explanation for your rating.

C#			
Criteria	*Definition of Criteria*	*Grade*	*Grade Explanation*
Efficiency	If a language is simple to read. If a user is a programmer, it should be simpler to pick up new languages that are written well. This meaning they can translate the language easier	B	C# is very similar to any basic OOP language. It is easy to use well and doesn't have unique syntax so someone who doesn't know C# can translate it easy.
Simplicity	The overall simplicity refers to three different things: 1. Number of basic constructs: larger number of different constructs can make a language more difficult to pick up. 2. Feature multiplicity: how many ways there are to do the same task. 3. Operator Overloading: when a operator can have more	B	Like before, the program is similar to many OOP languages so it is easy to read. There are a few extra features that can be learnt that may not make sense from the start (using statements, LINQ, etc..).

	than one meaning. If this is not done correctly, can make it difficult to understand. Also, it has to do with whether the program is easy to read and easy to write for a new programmer picking up the language.		
Orthogonality	Combinations of smaller sets of constructs and combining them in a small number of ways. To summarize, when a language is orthogonal this means you can run an operation and this operation will affect nothing outside of it. This is more directed to lower level languages such as Assembly (example: addition can affect registers).	A	It's hard to determine this for higher level languages like C#, but we can say it is since for the most part an operation will never affect anything outside of it.
Syntax Design	This is simply how the language is designed writing-wise. Things like spelling, form, meaning, words, etc..	A	Again, similar to many OOP languages so the syntax is basically the same.
Reliability	We can say that something is reliable it will perform its intended function under any condition provided. This included things like type checking, exception handling, aliasing, etc..	A	All functions perform as intended, I have never had problems with this.
Abstraction Facilities	This talks about a programming languages ability to take structures and re-use them and their operations for similar purposes. This reduces the amount of code that needs to be written. An example of this is classes.	A	Since C# us an OOP language, it has these features (classes, structs, etc…)

2. (2 marks). What is the disadvantage of having too many features in a language?

Having a language with many features means there are many things a language can do. Usually, a programmer will stumble upon these features as they require them and search for them. However, if a programming language has many features, the language itself can become really hard to read and understand. This means it is harder for someone outside the project to pick it up and understand what is happening in the code.

3. (2 marks).What does it mean for a program to be reliable?

Similar to the definition of reliability in a programming language, reliability is when something does what it is supposed to do. For a program, it means that it can perform the functions it is supposed to without flaw no matter what the input conditions are. This means the program can function under all conditions, not just the ideal condition.

4. (3 marks).What is an example of two language design criteria that are in direct conflict with each other?

A good example is the flexibility of a program, also similar to writability, and the reliability of the program itself. A programming language can include lots of flexible features (pointers in C and C++, or directly managing memory in Assembly), however that means the languages may be less reliable. These extra features may not always act as the user intended causing errors. C# could be considered a reliable language, but one does not have the same flexibility because it is missing lots of features that C++ or C has.

5. (3 marks).Why is the von Neumann bottleneck important?

The Von Neumann bottleneck is known as a bottleneck that is related to the speed of the connection to the computer vs. how fats a program can run. This relates to the connection speed between the memory of a computer and processor. However, program instruction can usual run faster than this computer connection speed, which creates the ‘bottleneck’. This became the primary reason people started to develop what is know as parallel computers.

Chapter 2

1. (30 marks). Compare between two programming languages of your choice using the features listed below:

a. Object orientation

b. Static /Dynamic typing

c. Overloading

d. Functions as first class entities

e. Garbage Collection

f. Pointer Arithmetic

1. Describe briefly each feature

2. Do the comparison in a table format

Features	*Description*	*C*	*C#*
Object Orientation	The ability of a language to make its own objects such as classes, structs, data types, ect..	C cannot do classes.	C# can do all of of that.
Static/Dynamic Typing	Static typing is when the type of the variable is known at compile time, or is defined before compiling. Dynamic typing means that the type is known at runtime and is not predetermined/typed.	Statically typed.	Also statically typed.
Functions as First Class Entities	This means we are treating functions as variables; allowing them to be assigned to variables, sending them into another functions, etc..	Supported for high level functions, but not nested or lower.	Yes it can with C# 11.

Garbage Collection	This means that variables and other objects that are no-longer in use are automatically disposed of to not take up anymore memory.	Doesn't have garbage collection, need to program this in manually	Has automatic garbage collection
Overloading	Operators can have different meanings depending on its use and implementation.	C does not support this.	C# can do this.
Pointer Arithmetic	Ability of a programming language to use pointers.	C does support pointer arithmetic.	C# does not include pointers at all.

2. (2 marks).Why was the slowness of interpretation of programs acceptable in the early 1950s?

In the early 50's, math programs were mean to be coded in mathematical notation, where as programs for data processing were supposed to be written in English. The main reason for the slowness was the lack of interpretation, a translator had not yet been implemented at the time so it was slow. Also, for the two different types of programming companies almost needed two separate computers to process the different languages. Another reason that could be the cause was lack of floating point numbers, or lack of different data types to support different operations (such as arrays).

3. (2 marks).What was the primary application area of computers at the time Fortran was designed?

Around that time, computer applications were only used for doing scientific and mathematical programs. Basically, programs were made only to do math equations for the user. There was no programming methodology and they were usually small programs.

4. (2 marks).What three concepts are the basis for object-oriented programming?

The three main concepts of OOP are data abstraction, inheritance, and dynamic binding.

5. (2 marks). LISP began as a pure functional language but gradually acquired more and more imperative features. Why?

LISP was mainly used as a programming language for artificial intelligence. After some time, they developed Scheme and Common Lisp which started to incorporate some imperative programming features. This was probably because of the development of competing languages and the need for LISP to adapt to recent requirements for AI and other applications.

6. (2 marks). Explain two reasons why pure interpretation is an acceptable implementation method for several recent scripting languages.

Scripting languages interpret and execute one command at a time. The first reason is because it is easier to type and easier to learn. All variables are dynamically typed so the user doesn't worry about datatypes as much as they are evaluated at runtime. The second is because the language is usually run on a server that runs web pages dynamically.

Programming Exercise

2. To understand the value of recursion in a programming language, write a program that implements quicksort, first using recursion and then without recursion.

-- Code In Folder --

Also on github:
https://github.com/herasymj/Fall2018_Class_Assignments/tree/master/CS_350/Assignment1