Review Test Submission: Quiz 3 - Programming Paradigms

User	Kaidi He
Course	201503_Advanced Software Paradigms_CSCI_6221_11
Test	Quiz 3 - Programming Paradigms
Started	9/23/15 6:15 PM
Submitted	9/23/15 6:40 PM LATE
Due Date	9/23/15 6:40 PM
Status	Completed
Attempt Score	85 out of 100 points
Time Elapsed	24 minutes out of 30 minutes
Results Displayed	All Answers, Feedback, Incorrectly Answered Questions

Question 1

0 out of 15 points

✗ Lisp began as a pure functional language but gradually acquired more and more imperative features. The main reason why imperative features were put in Lisp was to increase its execution efficiency.

Answers: True
False

Question 2

15 out of 15 points

✓ The main motivation for the development of PL/I was to provide a single tool for computer centers that must support both scientific and commercial applications.

Answers: True

False

Question 3

20 out of 20 points

✓ Why, in your opinion, do new scripting languages appear more frequently than new compiled languages?

Response Feedback: [None Given]

Question 4

15 out of 15 points

✓ Regarding the evolution of programming langagues

Answers: Smalltak is frequently considered as one of the first programming language that fully supported object-oriented programming.

Lisp is one of the first functional programming languages. Lisp has several decendants, such as Scheme, Commom Lisp. Most of the functional programming constructors created by Lisp are also available in Ada and Algol. This is why both of them are also considered functional programming languages.

C# (aka C Sharp) was created by Bjarne Sroustrup at Bell Labs. It is based on Java and Visual Basic.

Both Java and JavaScript are a good examples of scripting languages

Response Feedback: Smalltak is frequently considered as one of the first programming language that fully supported object-oriented programming.

Question 5

20 out of 20 points

✓ What are the arguments both for and against the idea of a typeless language?

Response Feedback: [None Given]

Question 6

15 out of 15 points

✓ Regarding typeless languages, what statements are correct.

Answers: The syntax of typeless languages can be viewed as overly long or verbose when compared to typed languages.

One of the drawbacks of typeless languages is that type checking is impossible, so that it is entirely the programmer's responsibility to insure that expressions and assignments are correct.

The argument for typeless languages is their great flexibility for the programmer. Literally any storage location can be used to store any type value.

Typeless languages aren't as flexible as typed languages, as data structures need to be cast to the correct type before another object can receive them

Wednesday, October 7, 2015 6:47:52 PM EDT

← **OK**