

Take home Quiz-03

Due Feb 11 at 11:59pm**Points** 15**Questions** 15**Available** Feb 10 at 12:15pm - Feb 11 at 11:59pm 1 day**Time Limit** None

This quiz was locked Feb 11 at 11:59pm.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	16 minutes	8 out of 15

Score for this quiz: **8** out of 15

Submitted Feb 11 at 9:44am

This attempt took 16 minutes.

Question 1

1 / 1 pts

1) The *Just-in-Time Compilation* (JIT) has the ability to skip over code sections that will not be executed at run time, speeding up the program execution (interpretation).

Correct!☒ True☐ False

Question 2

1 / 1 pts

2) Early FORTRAN *implicit* name declarations introduced a **security loophole** in the language.

Correct!☒ True☐ False**Question 3****1 / 1 pts**

3) Early FORTRAN is a pioneer HLL in the introduction of a "true" ***built-in*** Abstract Data Types (ADTs).

Correct!☒ True☐ False**Question 4****1 / 1 pts**

4) Early FORTRAN is a pioneer HLL in the introduction of ***overworking*** the INTEGER type.

Correct!☒ True☐ False**Question 5****0 / 1 pts**

5) The major design goal of the early FORTRAN is ***Efficiency*** of programs' coding, while sacrificing speed of their execution time.

You Answered

☒ True

Correct Answer

☐ False

Question 6

0 / 1 pts

6) **Dynamic** type checking is inherently **insecure**, whereas the static type checking is much more **secure**.

You Answered

☒ True

Correct Answer

☐ False

Question 7

1 / 1 pts

7) A language with a **static** checking typing system is more **powerful** than a language with **dynamic** checking typing system.

☐ True

Correct!

☒ False

Question 8

0 / 1 pts

8) In early FORTRAN, there was no **global** names declarations.

You Answered

☒ True

Correct Answer

☐ False

Question 9

0 / 1 pts

9) For gaining efficiency, early FORTRAN sacrificed all HLLs features, e.g., power, abstraction, security, etc, for gaining **efficiency**. .

You Answered

☒ True

Correct Answer

☐ False

Question 10

1 / 1 pts

10) The use of **by-value-result** parameter passing in the early FORTRAN versions was to alleviate/solve some of the problems introduced by the use of **by-reference**.

Correct!

☒ True

☐ False

Question 11

0 / 1 pts

11) The “GO-TO label” statement is a security loophole in early FORTRAN versions.

You Answered

☒ True

Correct Answer

☐ False

Question 12

0 / 1 pts

12) It is enough for a HLL to be **secure** if it has a **rich set of types** and **secure typing system**.

You Answered

☒ True

Correct Answer

☐ False

Question 13

0 / 1 pts

13) The typing mechanism in HLLs is introduced to gain language **security** via checking the formal actual parameters matching both numbers and types.

You Answered

☒ True

Correct Answer

☐ False

Question 14

1 / 1 pts

14) Early FORTRAN had a ***rich*** and ***strong typing system*** that is why NASA used it.

☐ True☒ False

Correct!

Question 15

1 / 1 pts

15) When a general purpose HLL designer ***trusts*** the programmer to write ***secure*** code, ***without verifying*** the code for programmer's errors (via the compiler/run-time line of defenses), then such HLL is ***insecure***.

☒ True☐ False

Correct!

Quiz Score: 8 out of 15