

CS3723 HOMEWORK #10, Due: 2pm, 5/29/08

Note: If you collaborated with your classmates or used their notes, please note which classmates you collaborated with. If you use an external source, besides the text book, lectures, notes provided by the instructor, and your own intellect, please cite that source. Use quote marks if you are quoting material word-for-word from any source (including the text book).

C++ (Chapter 12)

1. (a) List the 4 main design goals of C++.
(b) How are these similar and/or different from the goals underlying the design of Simula and SmallTalk?
2. (a) Why did Bjarne Stroustrup choose not to use garbage collection for C++?
(b) Where did C++'s memory model come from?
(c) What C entity are C++ objects a generalization of?
3. (a) What was the most successful programming language of the decade from the mid-1980's to mid-1990's?
(b) Describe the general design philosophy behind C++. Does it try to enforce a particular style of or approach to programming?
4. (a) List 5 non-object-oriented enhancements to C++ relative to C.
(b) Describe two of them.
5. Explain one problem (related to subtyping) with how objects allocated on the stack.
6. (a) Give an example of the syntax for creating a derived class in C++
(b) Which of the four object-oriented concepts is provided by 'derived' classes in C++?
(c) Explain what is the difference between the public and private base classes of a derived class?
7. (a) Which of the four object-oriented concepts is provided by 'virtual functions' in C++?
(b) What is the name of data structure used to facilitate this?
(c) Describe how a virtual function call is carried out in C++?
(d) How would you compare this to messages in SmallTalk? What is the asymptotic complexity in terms of the height of the class hierarchy (i.e., the longest chain of classes that inherit from other classes)?
(e) What differences in design goals of C++ and SmallTalk do you think lead to these differences in mechanisms? Why?
8. (a) What are some of the problems with multiple-inheritance?
(b) How does C++ solve the problem with name clashes?
(c) What are virtual base classes?
(d) How does C++ represent classes with diamond inheritance if virtual base classes are not used?
9. Do Exercise 12.1 in Mitchell (Assignment and Derived Classes).
10. Do Exercise 12.6 in Mitchell (Subtyping and Visibility).

JAVA (Chapter 11)

11. (a) List seven design goals for the Java programming language.
(b) How do these goals and their priority in the design of Java compare to those for C++? (Which of these goals were not high-priority goals for C++? Which of these goals are of a lower priority for Java than for C++? Which goals for C++ are not goals for Java?)
(c) How do these goals and their priority compare to those of SmallTalk?
12. In what ways is Java simpler than C++?
13. What is *this/self* in C++, Java, and SmallTalk?
14. Early implementations of the Java Virtual Machine relied on interpretation (as described on the text book). Modern implementations rely on just-in-time compilation (especially for methods that are called many time and methods containing loops that repeat many times). What design goal of Java does just-in-time compilation further?
15. Why was Java designed to use a virtual machine? (What benefits does this provide?)
16. (a) Why are Java programs immune to the buffer overflow attacks that are endemic in C programs?
(b) What is the role of the byte code verifier? Give a scenario under which it would reject a class that is being loaded.
17. (a) What is the Java keyword associated with inheritance?
(b) Does Java support multiple inheritance?
(c) If you want to create a class that is a subtype of multiple 'entities,' ...
 - i. What kind of 'entities' do you have to have? (Hint: not 'class')
 - ii. How are these entities different from classes?
 - iii. What Java keyword is associated with this subtyping?
18. There are four different bytecodes for invoking a method.
 - (a) What are they?
 - (b) Explain what three of the four are used for?
 - (c) Describe the steps involved in `invokevirtual` (assume it has been rewritten to use an offset in what Mitchell calls the "mtable")?
 - (d)
19. Do Exercise 13.6 in Mitchell (Java Bytecode). (Hint: Think about Exercise 3.5.)
20. Do Exercise 13.7 in Mitchell (Exceptions, Memory Management, and Concurrency).