CS365 Final
Spring 2020

I state that this is my work, and only my work, and that I did not receive any assistance, in any form, from anyone else, student or otherwise. I understand that failure to comply with this statement may result in an 'F' for the class and that I may be reported to the Dean of Students for academic dishonestly.

Name _____ **Date** _____

10 points each. You need to answer 10 of the following questions. You must place an X in the response area for the question you choose not to answer.

1. What are the 2 most common means of passing arguments to parameters discussed in class? List strength and weakness of each of both approaches.

2. Where the previous question asked about *how* an argument is passed to a parameter, in this question list several approaches different languages use to create "flexibility" regarding *what*, or *how many* arguments are passed to the method. Your answer will include approaches from more than one language.

3. What are the goals of object-oriented programming? Describe each goal, and how it (supposedly) improves the software development process.

4. You cannot create objects from an interface. What is an interface, and why use an interface if you cannot create an object from it?

5. What is a "referentially transparent function?" What is programming in a "purely functional style"? How is programming in a purely function style achieved?

6. Most languages overload at least one operator. Some languages, such as C++, allow for user-defined overloaded operators, while other languages, such Java, do not allow this to happen. What are the advantages and disadvantages of allowing operators to be overloaded?

7. An `ArrayList` provides a convenient means of creating an "array like" data structure that can grow and shrink, yet provide "array like" direct access to the data through `set` and `get` methods. What we know about how arrays work tells us that the array cannot just grow or shrink as needed. Why would the `ArrayList` still need to have a basic array structure and why can't it really grow or shrink?

8. What type of information goes into an activation record or a stack frame? When does the system create an activation record? When does the system release, or free, and activation record?

9. What is the difference between overloading and overriding, and when would you use each?

10. Define "strongly-typed" and "weakly-typed" languages. Give an example of a strongly typed language. Describe several advantages of a strongly typed language.

11. Some languages include a return value (including void) in the method signature, while others do not. What is the method signature? What are the differences in the languages that allow for this distinction? What are the advantages and disadvantages of these approaches?