PROGRAMMING AND ALGORITHMS II

WEEK 2. ABSTRACTION AND ANALYSIS

QUIZ

1. To use functions/classes/methods defined in a library correctly, you must understand the API (that is, what the parameters and return values are). **TRUE**
2. Assuming the preconditions and postconditions and code are correct, the postcondition is guaranteed to be true after the code is executed if the precondition is met before the code is executed. **TRUE**
3. A function that detects a violation of its precondition should print out an error message. **FALSE**
4. A function's signature provides a complete specification of its behavior. **FALSE**
5. A well-designed function/method often has undocumented side effects. **FALSE**
6. Using the same computer, programming language, and input data, executing an algorithm that is Theta(n) must be faster than executing an algorithm that is Theta(n^2). **FALSE**
7. A function with more lines of code can be faster than a function with fewer lines. **TRUE**
8. Theta notation is an effective measure of algorithm efficiency when the expected input size for the algorithm is small. **FALSE**
9. All O(n^2) algorithms are Theta(n^2) **FALSE**
10. All Theta(n^2) algorithms are O(n^2). **TRUE**

MULTIPLE CHOICE QUESTIONS

1. Which of the following is not part of the signature of a function?
    a. The name of the function
    b. **how the function works**
    c. the parameters
    d. the return value

2. Which of these actions inside a function would produce a side effect?
    a. setting an immutable parameter to a new object
    b. setting a mutable parameter to a new object
    c. **modifying a mutable parameter**
    d. returning a value

3. Which of the following indicates that a function's precondition was met?
    a. the function does not crash
    b. the function returns a value
    c. the function raises an exception
    d. **none of the above**

4. In general what will have the biggest effect on how long your algorithm takes to execute on a large data set?
    a. **the efficiency of your algorithm**
    b. the computer language used to implement the algorithm
    c. the number of lines of code in your algorithm
    d. the speed of the hard disk on the computer

5. A function with two loops has an asymptotic running time of
    a. 0(log2n)
    b. 0(n)
    c. 0(n2)
    d. **not enough information to determine**

6. If a 0(n^2) algorithm requires 3 seconds to execute on an input of one million elements, approximately how long should it take on an input of two million elements?
    a. 6 seconds
    b. 9 seconds
    c. 12 seconds
    d. 18 seconds

7. If a 0(n^3) algorithm requires 4 seconds to execute on an input of one million elements, approximately how long should it take on an input of two million elements?
    a. 8 seconds
    b. 16 seconds
    c. 32 seconds
    d. 64 seconds

8. If a $0(log_2 n)$ algorithm requires 20 seconds to execute on an input of one million elements, approximately how long should it take on an input of two million elements?
    a. 21 seconds
    b. 25 seconds
    c. 30 seconds
    d. 40 seconds

9. If a 0(2^n) algorithm requires 10 seconds to execute on an input of 10 elements, approximately how long should it take on an input of 20 elements?
    a. 20 seconds
    b. 100 seconds
    c. 1,000 seconds
    d. 10,000 seconds

10. If a computer is capable of performing one billion operations per second, approximately how long would it take to execute an algorithm that requires n^2 operations on an input of two million elements.
    a. 400 seconds
    b. 2,000 seconds
    c. 4,000 seconds
    d. 20,000 seconds

WEEK 3. DATA ABSTRACTION

QUIZ

1. To implement an ADT in Python, you must use classes. **FALSE**
2. If the programming language supports classes, you should usually use them when implementing an ADT. **TRUE**
3. Class variables can be shared by all instances of a class. **TRUE**
4. When designing a program, one way of locating potential objects is by looking for verbs in the system description. **FALSE**
5. Encapsulation refers to combining the data and methods into one syntactic unit. **TRUE**
6. With polymorphism, a programmer writes multi-way if statements to check the type of an object and determine which method to call. **FALSE**
7. Subclasses inherit methods defined in their superclasses. **TRUE**
8. Operator overloading allows programs to compute results that could not be computed without operator overloading. **FALSE**
9. To do operator overloading in Python, you must use classes. **TRUE**
10. Unit tests should be executed whenever you make a change to a class. **TRUE**

MULTIPLE CHOICE

1. When developing large software systems, you should:
    a. immediately sit down at the computer and start writing code
    b. design some of the system, write some code, possibly redesign it, and test the components as you write them
    c. design the entire system before you write any code
    d. implement the entire system before you test any of the code

2. Which parts of the program description will be most helpful in identifying possible objects for a system design?
    a. adjectives
    b. nouns
    c. verbs
    d. all of the above

3. Which parts of the program description will be most helpful in identifying possible methods in a system design?
    a. adjectives
    b. nouns
    c. verbs
    d. all of the above

4. How do you distinguish between instance variables and local variables for a method?
    a. instance variables are part of the data for a particular object and are needed in multiple methods while local variables are needed only within that method
    b. a class should never use local variables; all variables used in methods should be instance variables
    c. a class should never use instance variables; all variables used in methods should be local variables
    d. instance variables should be used for constants only

5. If you are examining a Python class that someone else wrote, how do you determine if a variable is a local variable or an instance variable?
    a. the same variable name is used in more than one method
    b. the variable is accessed by placing self . before the variable name
    c. the variable is used in the __ init__ method
    d. instance variables are always preceded by an underscore

6. When should you use class variables?
    a. when each instance of the class needs its own copy of the data
    b. when each instance of the class can share the same copy of the data
    c. when the data is constant
    d. b and c

7. If you are designing a class to represent a polynomial, which of the following should be instance variables?
    a. the coefficients
    b. a value to evaluate with the polynomial
    c. the result of evaluating the polynomial with a specific value
    d. all of the above

8. If you are designing a class to represent a polynomial, which of the following should be class variables?
    a. the coefficients
    b. a value to evaluate with the polynomial
    c. the result of evaluating the polynomial with a specific value
    d. none of the above

9. When writing unit tests using the Python unittest framework the test code is written as
    a. a number of functions
    b. a separate class that subclasses your class
    c. a separate class that subclasses unittest . TestCase
    d. part of the class you are testing

10. What is the purpose of unit testing?
    a. to help you to think about your design
    b. to help you find errors in your code
    c. to allow you to easily test your code each time you change it
    d. **all of the above**

WEEK 4. CONTAINER CLASSES

QUIZ

1. Python is the only high-level language that has a built-in container type for sequential collections. **FALSE**
2. The indexing operation on lists returns a sublist of the original. **FALSE**
3. Python lists are implemented using contiguous arrays. **TRUE**
4. A Python list is a homogeneous container. **FALSE**
5. Arrays do not allow efficient random access **FALSE**
6. On average, appending to the end of a Python list is a Theta(n) operation. **FALSE**
7. Inserting into the middle of Python list is a Theta(n) operation. **TRUE**
8. Python is unique in that it has a built-in container type that implements a general mapping (dictionaries). **FALSE**
9. Python dictionary keys must be immutable objects. **TRUE**
10. Looking up an item in a Python dictionary is a Theta(n) operation. **FALSE**

MULTIPLE CHOICE

1. Which of the following is not true of Python lists?
    a. They are implemented underneath as contiguous arrays.
    b. **All of the items in a list must be of the same type**.
    c. They can grow and shrink dynamically.
    d. They allow for efficient random access.
2. Which of the following is a 8(n) operation?
    a. Appending to the end of a Python list.
    b. Sorting a list with selection sort.
    c. **Deleting an item from the middle of a Python list.**
    d. Finding the ith item in a Python list.
3. Which of the following is not a method of the Deck class presented in the chapter?
    a. size
    b. shuffle
    c. deal
    d. **All of the above are methods of the class.**
4. Which of the following is not a method of the Hand class presented in the chapter?
    a. add
    b. sort
    c. deal
    d. All of the above are methods of the class.
5. What is the time efficiency of the selection sort algorithm?
    a. 0(logn)
    b. 0(nlogn)
    c. 0(n)
    d. 0(n^2)

6. What is the time efficiency of the Python built-in list method sort?
    a. 0(logn)
    b. **0(nlogn)**
    c. 0(n)
    d. 8(n^2)
7. What is the time efficiency of the operation max (myList) ?
    a. 0(logn)
    b. 0(nlogn)
    c. **0(n)**
    d. 0(n^2)
8. What operation is not supported for Python dictionaries?
    a. Item insertion
    b. Item deletion
    c. Item lookup
    d. I**tem ordering (sorting)**
9. Which of the following is not true of Python dictionaries?
    a. They are implemented as hash tables.
    b. **Values must be immutable.**
    c. Lookup is very efficient.
    d. All of the above are true.
10. A trigram model of natural language
    a. uses a prefix of three words to predict the next word.
    b. **uses a prefix of two words to predict the next word.**
    c. is more useful than a Markov model.
    d. is used to send money overseas.

WEEK 6. C++ INTRODUCTION I

QUIZ

1. All C++ programs must have a function named main. **TRUE**
2. Any variable used in a C++ program must be declared with a type before it can be used. **TRUE**
3. A C++ function must return a value. **FALSE**
4. A C++ program that compiles will output the results that you intend it to. **FALSE**
5. If the C++ compiler outputs a warning, it will never compile the program. **FALSE**

MULTIPLE CHOICE

1. Which of the following programs would you expect to be significantly faster when written in C++ than when written in Python?
    a. a program to convert miles to kilometers
    b. a program with a loop that runs a million times
    c. a program with a loop that runs 10 times
    d. all of the above
1. If a C++ function uses a variable that has not been declared, what happens?
    a. The code will not compile
    b. When executing that function, an error message will be generated similar to Python's NameError message.
    c. The program crashes.
    d. none of the above
3. Compiling a C++ file that does not contain a main function produces
    a. an executable program.
    b. an object file containing the machine code for that C++ file.
    c. another C++ file
    d. none of the above

4. The linker
    a. copies header files into a C++ file.
    b. compiles a C++ file into machine code
    c. combines machine code from multiple files to produce an executable program.
    d. loads a program into memory and executes it.
5. C + + functions can return
    a. at most one variable or expression.
    b. multiple variables or expressions.
    c. arrays.
    d. C++ functions do not return a value.
6. Which of the following statements is the most similar in Python and C++?
    a. the for statement
    b. a function definition
    c. the if/else statement
    d. **the while statement**
7. Which of the following is not true about the C++ pass by reference mechanism?
    a. All changes to the formal parameter that are made in the function affect the actual parameter.
    b. A copy of the actual parameter is made.
    c. It allows you to effectively return multiple values calculated by the function.
    d. It is slower than pass by value.
8. Which of the following is true about C++ arrays?
    a. Arrays can be passed by value.
    b. Arrays include a method to sort the values in the array.
    c. The values in the array must be the same type.
    d. Arrays can be returned by a function.
9. What is the main purpose of a header file?
    a. to comment the code in a source file
    b. to declare items so they can be used in C++ source files
    c. to define iterns so they can be used in C++ source files
    d. none of the above
10. The scope of a variable refers to
    a. the different values it can hold
    b. **where the variable can be accessed**.
    c. the time during which memory is allocated for the variable.
    d. the name of the variable.

WEEK 7. C++ INTRODUCTION II

QUIZ

1. If you compile a C++ program using the Linux operating system on an Intel chip, you can execute the generated program on a computer running the Windows operating system on the same Intel chip. **FALSE**
2. For simple text-based programs you can usually recompile a C++ program on different architectures and operating systems without changing your code. **TRUE**
3. In general, a compiled C++ program will execute faster than a similar Python program on the same computer. **TRUE**
4. A C++ program that solves a specific problem such as sorting numbers will always execute faster on the same computer than a Python program that solves the same problem. **FALSE**
5. Passing an int type by reference is faster and more efficient than passing an int type by value. **FALSE**
6. A C++ program that does not compile can be executed. **FALSE**
7. C++ compiler warnings should be ignored. **FALSE**

WEEK 8. CLASSES

QUIZ

1. C++ classes have a constructor that has the same name as the class. **TRUE**
2. C++ constructors are called automatically. **TRUE**
3. You must write code for the C++ constructor of every class you write. **FALSE**
4. Methods of a C++ class can create or add additional instance variables to the class. **FALSE**
5. Methods may be declared in the private section of the class definition. **TRUE**
6. Instance variables must be declared in the private section of the class definition. **FALSE**
7. A compiler error is generated if a method has a variable with the same name as an instance variable. **FALSE**
8. Methods may be written inline in the header file. **TRUE**
9. A class method can access instance variables. **FALSE**
10. A method can access both class variables and instance variables. **TRUE**

MULTIPLE CHOICE

1. In C++, instance variables may be declared
    a. private only
    b. public only
    c. protected only.
    d. public, private, or protected.

2. In C++, instance methods may be declared
    a. private only.
    b. public only.
    c. protected only.
    d. public, private, or protected.

3. Members of a class that are declared private may be accessed
    a. only by methods of the class.
    b. only by methods of the class or friends of the class.
    c. only by methods of the class, subclasses of the class, or friends of the class.
    d. by any code.

4. Members of a class that are declared protected may be accessed
    a. only by methods of the class
    b. only by methods of the class or friends of the class.
    c. only by methods of the class, subclasses of the class, or friends of the class.
    d. by any code.

5. Members of a class that are declared public may be accessed
    a. only by methods of the class
    b. only by methods of the class or friends of the class.
    c. only by methods of the class, subclass, or friends of the class.
    d. by any code.

6. Methods that are declared const
    a. declare constants within the method.
    b. cannot modify any of the instance variables
    c. must have parameters that are all const.
    d. must return a constant.

7. If you are examining a C++ class that someone else wrote, how do you determine if a variable is a local variable or an instance variable?

a. The same variable name is used in more than one method.
b. The variable is used in the constructor.
c. Instance variables are always preceded by an underscore.
d. Instance variables are declared within the class definition, not in one of the methods.

8. How can C++ operators be written?

a. They can only be written as members of a class.
b. They can only be written as functions.
c. They can be written as either members of a class or functions.
d. Some can only be written as functions while many can be written as functions or methods.

9. Where are C++ class variables accessible?

a. Their access depends on whether they are declared private, protected, or public
b. They are accessible only by the methods in the class.
c. They are accessible only by class methods.
d. They are accessible anywhere.

10. C++ class variables are declared by

a. using the keyword class before the variable type.
b. using the keyword static before the variable type.
c. putting them in the header file, but after the ending brace for the class.
d. declaring them inside the constructor.

WEEK 9. TEMPLATES

QUIZ

1. Templates allow you to write code once and reuse it with multiple types. **TRUE**
2. The compiler will always catch syntax mistakes in your C++ template functions and methods. **FALSE**
3. For each data type a template function is called with, the compiler generates a separate copy of the machine language instructions for the function. **TRUE**
4. You can place template function or method implementations in an implementation file (.cpp) and the linker will correctly link the code so it can be called from other implementation files **FALSE**
5. Templates give you the same flexibility that Python's dynamic typing does. **FALSE**

MULTIPLE CHOICE

1. When you write a template function,
   A. the compiler generates one set of machine language instructions for all types.
   B. the compiler generates a separate set of machine language instructions for each type that you call the template function with.
   C. the compiler generates a separate set of machine language instructions for every built-in type and every class your program uses whether or not the template function is called with each type.
   D. the C++ run-time environment generates the machine language instructions as needed when the function is called with different types.

2. What is/are the advantages of using templates instead of a typedef statement and cutting and pasting the code?
    A. The resulting executable program requires less memory.
    B. The resulting executable program will run faster.
    C. You do not have to write as much code or risk making errors when copying the code.
    D. all of the above
3. Which of the following are techniques for writing C++ template classes?
    A. You may write a class header file as you usually do and at the bottom of the file, include the file containing the implementation of the template methods
    B. You may write a class header as you usually do and write the implementation of the methods with the inline keyword.
    C. You may write a class header as you usually do and write the implementation of the methods without the inline keyword.
    D. a and b
4. Using a template class even when your program only creates an instance of the class with one data type
    A. Requires less memory than not using templates if you call all the methods.
    B. requires more memory than not using templates if you call all the methods.
    C. requires the same amount of memory than not using templates if you call all the methods.
    D. will execute more slowly than not using templates.
5. Based on the example using the vector class, what does the iter variable correspond to?
    A. the address of the iv variable
    B. the address of the current element in the array
    C. the value of the current element in the array
    D. none of the above