University of Bath

DEPARTMENT OF COMPUTER SCIENCE EXAMINATION

CM10227: PROGRAMMING 1

Thursday, 27 January 2011, 13.00–15.00

No calculators may be brought in or used.

Full marks will be given for correct answers to THREE questions. If you opt to answer more than the specified number of questions, you should clearly identify which of your answers you wish to have marked. In cases where you have failed to identify the correct number of answers the marker is only obliged to consider the answers in the order they appear up to the number of answers required.

1. (a) What are object and class diagrams? [4]

   (b) In Java, given two boolean expressions A and B, Is the following statement true or false? Briefly explain your answer.

   A & B is the same as B & A for any Boolean expressions A and B.

   [3]

   (c) Explain how selection sort works and discuss its time and space complexity. [6]

   (d) In software engineering, solutions to common design problems are often called design patterns. One of them is called Singleton and it describes how one can make sure that a certain class can have only one instance during a program's life-cycle. Further requests for objects will result in the same instance being returned. Implement a Singleton class. In the main method write a test to demonstrate that your code works correctly (i.e. only one instance is created even if more are requested). [7]

2. (a) Explain the following statement: Java is not completely object-oriented. [2]

   (b) Explain the concept of inheritance. How is this implemented in Java? [3]

   (c) We discussed two ways of differentiating programming languages on the basis of data types. Explain both and mention in which categories you place Python and Java [4]

   (d) Implement a Python version of the in-place (i.e. no slicing) quick-sort algorithm. [11]

3. (a) Explain how the call stack in Python works. [4]

(b) Explain aliasing in the context of variables. Provide an example of variables that are aliases and provide an example where two variables are referring to the same value but are not aliases. [4]

(c) Explain the concept of polymorphism. [3]

(d) A deque ("double-ended queue") is like a queue, except that it allows access at both ends. Create a Java class Deque which implements a deque. It should implement the following interface:

```
public interface DequeInterface {
        public void addFront(Object o);
                // adds item o to the front of the deque
        public void addRear(Object o);
                // adds item o to the rear of the deque
        public Object removeFront();
                // removes the item at the front and returns it
        public Object removeRear();
                // removes the item at the rear and returns it
        public int size();
                // returns the number of items which are currently
                // stored in the deque
}
```

[9]

4. (a) In terms of complexity, what does it mean to have a slow function? [3]

(b) How does method look up work in Java? Explain both compile and run-time. [5]

(c) Why is documentation important? [2]

(d) Consider the following piece of code:

```
def recurse(list1 ,list2 =[], list3 =[2 ,4 ,6]):
        if len(list1 )==0:
                print list2 * 3
                return
        el = 0
        t = 0
        for i in range(len(list1 )):
                list2 .append(list1 .pop(i) * list3[el] + i)
                el = (el + 1) % len(list3)
                t = t + recurse(list1 ,list2)
                list1 .insert(i,list2 .pop())
        return t

recurse([3 ,5],[2 ,1 ,3])
```

What is printed on the screen? What is the return value of the functions? [10]

5. (a) What are pure functions and how are they related to functional programming? [3]

(b) What is garbage collection? [3]

(c) Explain overloading both in terms of variables and methods. [4]

(d) Write a recursive **and** an iterative version of integer division. One version needs to be implemented in Java and the other one in Python. The function/method should be able to operate with positive and negative numbers.
You are **not** allowed to use the \ operator. [10]