Final Exam Information for V22.0101
Fall 2000

The exam takes place Tuesday December 19, at 2:00 p.m.:
- Overton’s section: Room 101, CIWW
- Levy’s section: Room 408, Main Building
- Marateck’s sections: Rooms UC50/UC52, Tisch Building

Topics covered: Chapters 1 through 9 and Chapter 11 of the textbook. There will be a question on recursion (Section 7.9). There will be special emphasis on arrays (Chapter 11) including two-dimensional arrays. You are not expected to know the Advanced Topics or the Random Facts in the text, and you are not expected to know about Vectors or Interfaces. The exam is CLOSED BOOK, WITHOUT NOTES. If you have difficulties with the sample questions here you can consult the TA, Naranjan, who will be holding office hours, Mon–Thurs, until Monday Dec 17 (see course web page for hours).

General instructions: write all your answers in the exam booklet provided. All questions will ask you to write methods, either static (class) methods, or instance (non-static) methods that access instance variables of a class.

None of the methods you write should read from the console or write to the output screen (in particular, there should be no calls to System.out in your solutions). None of the questions will involve applets. Read the questions carefully.

For questions involving strings, you may find the following methods useful:

- length (returns integer): if str is "hello", str.length() is 5
- charAt (returns char): if str is "hello", str.charAt(1) is 'e'
- subString (returns String): if str is "hello", str.substring(1,3) is "el"

You can expect about 6 questions on the exam. The following questions would be typical.

1. The Fibonacci numbers are the numbers 0, 1, 1, 2, 3, 5, 8, 13, 21, etc. (each number is obtained by adding the previous two together). Write a static method

   public static int firstFibPast(int n)

   that returns the first Fibonacci number that is bigger than or equal to the parameter n. For example, if n has the value 10, the method would return 13. If n has the value 0, the method would return 0. The method should not use recursion.

2. Write a static method

   public static int longestBlank(String str)

   that returns the maximum number of consecutive blanks in the String parameter str. For example, if the string is "abc def geh ijk", the method would return 2. If there are no blanks in the string, it returns 0.
3. Write a static method

   public static int toInteger(String str)

that converts the string given by the parameter str to a corresponding integer. For example, if str is the string "1234", the method returns the integer 1234. If the string contains any non-digit characters, the method returns 0. You may not use the parseInt method of the Integer class, but you may use the length, charAt and subString methods of the String class (see above).

4. Write a static method

   public static String reverseString(String str)

that returns the reverse of the parameter str. For example, if parameter str is "hello", the method should return "olleh". There are two ways to do this. One uses recursion (in this case, start by writing the method so it works for the empty string ""). The other way uses a loop. You should be able to do both, since an exam question might specify which is wanted.

5. Write a static method

   public static boolean prime(int n)

that returns true if the parameter n is a prime number, and false otherwise. (A number n is prime if the only positive integer less than n that divides n exactly (with zero remainder) is 1.)

6. Write a static method

   public static int sumDivisors(int n)

that returns the sum of the positive integers less than the parameter n that divide into n exactly (with zero remainder). For example, if n has the value 8, the method returns the value 7 (1 + 2 + 4) and if if n has the value 9, the method returns the value 4 (1 + 3).

7. Write a static method

   public static String concat(String[] words)

that returns a single string consisting of all the strings in the array words concatenated together. For example, if the elements of the array words are "Hello", "this", "is", "a" and "test", the method should return the string "Hellothisisatest". Remember that the elements of the array are words[0] through words[k], where k is one less than words.length, the length of the array (no parentheses, since this is not a method call).

8. Write a static method

   public static char pickOne(char c1, char c2)

that returns either the character given by the parameter c1 or the character given by the parameter c2, chosen at random. Your method should either use the library method Math.random, which returns a double between 0 and 1, or, if you prefer, the Random class (see Section 6.6 of the text).
9. Consider the following class and its instance variables:

```java
public class Bingo {
    // methods go here......
    int size; // these are instance variables
    char [][] board;
}
```

Write a constructor for the class Bingo:

```java
public Bingo(int n, char c1, char c2)
```

The constructor should (a) initialize the instance variable `size` to the parameter `n` and (b) initialize the instance variable `board` to a two dimensional array of characters with `size` rows and `size` columns, with each character initialized to either the character given by the parameter `c1` or the character given by the parameter `c2`, each one chosen randomly by calling your method written to answer Question 8 (which you can assume is available as part of the Bingo class).

10. This question also refers to class Bingo. Write a method (an instance method)

```java
public boolean rowBingo(char c)
```

that returns `true` if `board` has at least one row where every element in the row is the character given by the parameter `c`. Otherwise it returns `false`. (The first index of the array is the row index.)

11. This question also refers to class Bingo. Write a method (an instance method)

```java
public boolean allSame()
```

that returns `true` if every element in the two dimensional array `board` is the same character (this could be any character). Otherwise it returns `false`.

12. This question also refers to class Bingo. Write a method (an instance method)

```java
public int edgeCount(char c)
```

that counts how many elements on the edge of `board` (in the first or last row or column) are the character given by the parameter `c`.

13. Consider the following class and its instance variables:

```java
public class Grades {
    // methods go here......
    int numberOfStudents; // these are instance variables
    int numberOfTests;
    String [] studentName;
    String [] testName;
    int [][] score;
    double [] studentAverage;
    double [] testAverage;
}
```
Assume that a constructor has constructed a Grades object, with data assigned so that
studentName is an array of student names, testName is an array of test names, and
score is a two dimensional array of student test scores, with score[i][j] being the
score that student i received on test j. Do not write such a constructor. Your job in
this question is to write a method (an instance method)

    public void figureAverages()

that assigns to studentAverage the average score for each student (averaged over all
tests), and assigns to testAverage the average score for each test (averaged over all
students), using the data in score (and ignoring the student and test names).

14. Consider the following class and its instance variables:

    public class Date
    { // methods go here.....
        int month;    // these are instance variables
        int day;
    }

Assume that a constructor has been written to construct objects of class Date, so that,
for example, an object with month set to 7 and day set to 4 refers to the date July 4th.
Write a method (an instance method)

    public boolean summer()

that returns true if the date given by the instance variables is a date that falls in
summer, i.e., between June 21 and September 21 inclusive. (June is represented by 6,
and September by 9.) If the date is not in this range, or the date is invalid, summer
should return false. Remember that July and August have 31 days, but June and
September have only 30 days.

15. This question also refers to class Date. Write a method (an instance method)

    public boolean equals(Object another)

for the Date class that overrides the equals method of the Object class (see p. 374 of
the text). This method returns true if the date given by the implicit parameter is the same
as the date given by the explicit parameter another, and false otherwise. The method
should use the instanceof operator to check that the object parameter another is
actually an object of the Date class, before checking that the dates are equal.

For other good sample exam questions, see Problems P11.1 through P11.7 on p. 472-473 of
the text.