KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Information and Computer Science Department

2012/2013 First Semester (Term 121)

ICS102: Introduction to Computing (2-3-3)

FINAL EXAM

Tuesday, January 1st 2013, 07:00 PM – 09:00 PM

120 MINUTES

Student Information

Name:	KEY
ID:	

Circle your section

Al-Sukairy	SM 9:00 – 9:50 am	SM 10:00 – 10:50 am
Al-Turki	SM 8:00 – 8:50 am	SM 12:50 – 01:40 pm

Question No.	Maximum Score	Score
01	20	
02	10	
03	10	
04	15	
05	15	
06	15	
07	15	
TOTAL	100	

Question 1: (20 points):

Choose the correct answer in the following questions:

1. Consider the following array:
   ```
   int myArray [] = {7,9,-3,6,1,-1};
   ```
   What is the value of
   ```
   myArray[myArray[0] % myArray[3]];
   ```
   a. 7
   b. 1
   c. 9
   d. 0

2. What is displayed by:
   ```
   System.out.println("1" + new Integer(2) + 3);
   ```
   a. The statement has an error and won't compile
   b. 6
   c. 123
   d. 15

3. Consider the following array:
   ```
   int myArray[][] = {{1,2},{5,6,7},{8,9,10,11}};
   ```
   What is the value of
   ```
   myArray[myArray.length-1][myArray.length];
   ```
   a. 7
   b. 10
   c. 11
   d. It will cause an `ArrayIndexOutOfBound` Exception.

4. The keyword `this` refers to
   a. instance variables
   b. local variables
   c. global variables
   d. the calling object

5. A static method is one that can be used with a _____.
   a. instance variable
   b. local variable
   c. global variable
   d. the class name

6. The correct syntax for accessing the length of an array named `numbers` is:
   a. `numbers.length()`
   b. `numbers.length`
   c. both a. and b. above
   d. None of the above

7. Consider the following code fragment:

```
class Test {
   public int x;
   public Test(){}
   public Test(int a){x = a;}
   public static void main(String[] args) {
      Test a = new Test(10);
      Test b = new Test();
      a = b;
      a.x = 30;
      System.out.println("a= " + a.x + " b= " + b.x);}}
```

The output will be:

a. a= 30 b= 0
b. a= 30 b= null
c. a= 20 b= 0
d. a= 30 b= 30

8. When the following code runs:

```
String str = "I-like-programming";
str = str.substring(str.length()/2) + str.indexOf("like");
```

`str` will have the value:

a. ogramming3
b. ogramming2
c. rogramming3
d. rogramming2

9. If the method:

```
public float aMethod(float a, float b)
```

is in a class, which of the following is legal overloading:

a. `public double aMethod(float a, float b)`
b. `public float aMethod(float a)`
c. `public float aMethod(float a2, float b2)`
d. `public int aMethod(float a, float b) throws Exception`

10. Consider the following:

```
double [][] a = {{1,2,3,4}, {5,6,7},{8}};
double [] b = a[2];
```

What is the size of array `b`:

a. A 2-Dimensional array of size 3 x 1
b. A 2 Dimensional array of size 1 x 3
c. A 1-Dimensional array of size 3
d. A 1-Dimensional array of size 1

Question 2: (10 points):

Find the output of the following Java programs:

a.

```
class Test {
   public static void main(String[] args){
      int x[]={1,2,3,4,5};
      atest(x,4);
      for (int i = 0; i < x.length; i++)
         System.out.println(x[i]);
   }
   static void atest(int a[], int n){
      int i, t;
      for(i = 0; i < n/2; i++){
         t = a[i];
         a[i] = a[n-i-1];
         a[n-i-1] = t;
      }
   }
}
```

OUTPUT

```
4
3
2
1
5
```

b.

```
class Test {
   public static void main(String[] args) {
      String str [] = {"Riyadh","Dhahran",
                       "Dammam","Jeddah"};
      char arr [] = atest(str);
      for(int i = 0; i < arr.length; i++)
         System.out.println(arr[i]);
   }
   public static char [] atest(String [] a){
      char c [] = new char [a.length];
      for(int i = 0; i < a.length; i++){
         c[i] = a[i].charAt(i);
      }
      return c;
   }
}
```

OUTPUT

```
R
h
m
d
```

Question 3: (10 points)

Discover and correct syntax and logical errors in the following code (rewrite only the lines which need to be corrected):

a.

```
// returns true if and only if the array a has no repeated elements
//
public static boolean noRepeatedElements(int []a)
{


     for(int i = 0; i < a.length; ++i)
     for(int i = 0; i < a.length-1; ++i)

          for(int j = 1; j < a.length; ++j)
          for(int j = i+1; j < a.length; ++j)

               if(a[i] = a[j])  return false;
               if(a[i] == a[j]) return false;

     return true;


}
```

b.

```
// returns the sum of the diagonal elements of a square matrix
//
public static int sumDiagonal(int [][] m)
{


     if(m.length  != m[0].length())   return 0;
     if(m.length  != m[0].length)   return 0;

     int sum;
     int sum = 0;

     for(i = 0; i < m.length; ++i)
     for(int i = 0; i < m.length; ++i)

          for(j = 0; j < m.length; ++j)
          for(int j = 0; j < m.length; ++j)

               if(i != j)
               if(i == j)

                    sum += m[i][j];


     return sum;
}
```

Question 4: (15 points)

Write a program (main method only) that prints the first `n` prime numbers, where `n` is a positive integer that is read from the user.

For example, if the user enters 8, the program prints out the first eight primes as follows:
`The first 8 primes are: 2, 3, 5, 7, 11, 13, 17, 19.`

Your program should work properly for any positive integer value `n`.

```
public static void main (String [] args) {
      Scanner kb = new Scanner(System.in);
      System.out.print("Enter a positive integer: ");
      int n = kb.nextInt();

      int num = 2;
      System.out.print("The first " + n + " primes are: ");
      while(n > 0) {
           if(isPrime(num)) {
                System.out.print(num);
                if(n > 1)
                     System.out.print(", ");
                else
                     System.out.print(".");
                --n;
           }
           ++num;
      }
}

public static boolean isPrime(int n) {
      for(int i = 2; i < n; ++i)
           if(n%i == 0)
                return false;

      return true;
}
```

Question 5: (15 points)

Write a program (main method only) that prompts the user to enter 10 integers and stores them in an array. It then tests whether the array is a palindrome or not.

A palindrome array is an array which reads the same in both directions (left-to-right & right-to-left). For example, the array {6,-3,15,15,-3,6} is a palindrome, while {5,3,0,4,2,-6} is not.

```
public static void main (String [] args) {
      int [] a = new int[10];
      Scanner kb = new Scanner(System.in);
      System.out.print("Enter 10 integers: ");
      for(int i = 0; i < a.length; i++)
           a[i] = kb.nextInt();

      boolean isPalindrome = true;
      int i = 0, j = a.length - 1;

      while(i < j){
           if(a[i] != a[j]){
                isPalindrome = false;
                break;
           }
           ++i; --j;
      }

      if (isPalindrome)
           System.out.print("The array is a palindrome.");
      else
           System.out.print("The array is NOT a palindrome.");
 }
```

Question 6: (15 points)

Write a static method that takes as input a 2D array of integers, and returns as a result the index of the column that has the largest sum among all columns. You may assume that columns have different sums.

For example, if the 2D array is

```
3    1    2    0
0    1    3    1
2    1    1    2
```

The result is the index value 2 (the third column has the largest sum of all other columns).

```
public static int maxColumnIndex(int a[][]) {

     int maxSum = 0;
     int maxSumIndex = 0;

     for(int i = 0; i < a.length; i++) {
          maxSum += a[i][0];
     }

     for(int j = 1; j < a[0].length; j++) {
          int sum = 0;
          for(int i = 0; i < a.length; i++)
               sum += a[i][j];
          if (sum > maxSum) {
               maxSum = sum;
               maxSumIndex = j;
          }
     }
     return maxSumIndex;
}
```

Question 7: (20 points)

Suppose that we want to define a class `Temperature` that represents a temperature reading in either Celsius or Fahrenheit. Complete the implementation of this class as instructed below.

```
class Temperature {
```

a. Define two instance variables: `temp`, which is the temperature value, and `scale`, which can be either the character 'c' (or 'C') for Celsius , or 'f' (or 'F') for Fahrenheit.

```
private double temp;
private char scale;
```

b. Define a no-argument constructor that initializes the temperature to zero degrees Celsius.

```
public Temperature() {
     temp = 0;
     scale = 'c';
}
```

c. Define a two-argument constructor that initializes both instance variables with values given as parameters. If the given value for `scale` is invalid (some character other than 'c', 'C', 'f', or 'F'), the constructor should default to a Celsius scale.

```
public Temperature(double temp, char scale) {
     this.temp = temp;
     if (scale == 'f' || scale == 'F')
          this.scale = scale;
     else
          this.scale = 'c';
}
```

d. Define a `toString` method that returns a string containing the current temperature value followed by the letter 'C' or 'F' depending on the current scale.

```
public String toString() {
     String str = temp + "";
     if (scale == 'f' || scale == 'F')
          str += "F";
     else
          str += "C";
     return str;
}
```

e. Define a method `getTempC` that returns the current temperature value in Celsius. Recall that if the current value is in Fahrenheit, then you can get the Celsius equivalent using the formula

$$\frac{5}{9}(FahrenheitTemp - 32)$$

```
public double getTempC() {
     double value = temp;
     if (scale == 'f' || scale == 'F')
          value = (5.0 / 9) * (temp - 32);
     return value;
}
```

f. Define an `equals` method that returns true if the current temperature object has a temperature value that is equivalent to the temperature value of the object given as a parameter (Notice here that the two temperatures can be in different scales).

```
public boolean equals(Temperature other){
     return this.getTempC() == other.getTempC();
}
```