INSTRUCTIONS:
100 marks

Questions 1-6, which are worth 10 marks each, are code-writing questions and should be answered in the spaces provided on the question paper.

Question 7, which is worth 40 marks, consists of 20 multiple choice parts. Each of these should be answered in pencil on the computer-readable multiple choice answer sheet supplied.

The exam is CLOSED BOOK — books and notes are not permitted. Students must return the exam question paper at the conclusion of the examination.

PLEASE NOTE

Examination candidates may only bring authorised materials into the examination room. If a supervisor finds, during the examination, that you have unauthorised material, in whatever form, in the vicinity of your desk or on your person, whether in the examination room or the toilets or en route to/from the toilets, the matter will be reported to the head of school and disciplinary action will normally be taken against you. This action may result in your being deprived of any credit for this examination or even, in some cases, for the whole unit. This will apply regardless of whether the material has been used at the time it is found.

Therefore, any candidate who has brought any unauthorised material whatsoever into the examination room should declare it to the supervisor immediately. Candidates who are uncertain whether any material is authorised should ask the supervisor for clarification.
Instructions

Read the entire paper first and plan your time. In general, you can get full marks on a code-writing question more easily than getting five multiple-choice questions correct, so you may wish to consider attempting the code-writing questions first.

Minor syntactic errors will not be punished in the code-writing questions.

The questions, including the multiple-choice questions, are not in any particular order of difficulty, so answer the ones that you find straightforward first.

Don’t forget to include your name and student number on the multiple-choice answer sheet, and double-check your student number as it will be read by computer.

DO NOT SPEND TOO MUCH TIME ON THE MULTIPLE CHOICE.
THEY SHOULD AVERAGE ONLY 2.6 MINUTES EACH.
1.

The Bahrain national flag is a rectangle with a width:height ratio of 2:1 and showing the following pattern:

```
           white
         \    /    \
          \  /     \
           \//      \
            /        \
           /         \
          /          \
         /           \
        /             \
       /               \
      /                 \
     /                   \
    /                     \
   /                       \
  /                         \
 /                           \
/                             \
```

Write a method

```java
public void drawBahrain(int n)
```

that draws the Bahrain flag on the screen with a height of n pixels. Your method should create and use a `SimpleCanvas` (as used in lectures and laboratories) to draw on. Both of the colours needed are pre-constructed `Color` objects.

(Do not worry about the precise positions and angles of the jagged bits: draw something approximately right.)

**Put your answer on the next page**
1st SEMESTER EXAMINATIONS
JAVA PROGRAMMING 1200

CITS.1200

Answer the Flag Question Here

(10 marks)
2.

(a) Write a method

```java
public int countSmaller(int x, int[] a)
```

that returns the number of elements of `a` that are smaller than `x`. (5 marks)
(b) Write a method

```java
public int median(int[] a)
```

that returns the median of `a`. The median of an array is the value `k` such that half of the elements in the array are smaller than `k`, and half are bigger than `k`. You may assume that `a` has odd length and that its elements are all distinct.

(5 marks)
3.

We wish to estimate the area of the intersection of two overlapping circles with diameters $d_1$ and $d_2$, whose centres are separated by $x$. We can do this by a simple sampling procedure.

Imagine the two circles are enclosed in a rectangle with width $d_1 + d_2$ and height $dk$, where $dk$ is the greater of $d_1$ and $d_2$:

![Circles and Rectangle](image)

Generate $n$ points randomly in this rectangle, and count how many points sit in the intersection (i.e. inside both circles). The proportion of points that sit in the intersection approximates $A/B$, where $A$ is the area of the intersection, and $B$ is the area of the rectangle. From this we can estimate $A$. Clearly the precision of the estimate increases with the number of samples.

Write a method

```java
public double area(double d1, double d2, double x, int n)
```

that implements the above procedure.

You should use an object from the library class `java.util.Random` to generate a sequence of random points inside the rectangle, and you will need the following method from that class.

```java
public double nextDouble()
```

Returns the next pseudorandom, uniformly-distributed double between 0.0 and 1.0 from this random number generator's sequence.
ANSWER THE CIRCLES QUESTION HERE

(10 marks)
Consider the class `BankAccount`, for use in a bank's account record system:

```java
public class BankAccount {

    // The accName is the username of the account holder
    // such as BillGates or JackSmith
    private String accName;
    private int accNumber;
    private int balance;

    public String getName()
    { return accName; }

    public int getBalance()
    { return balance; }

    // constructor and other details omitted
}
```

The records are sorted in alphabetical order by the account names. Write a method

```java
public int accountBalance(BankAccount[] acclist, String accName)
```

that uses an efficient search technique to search the sorted account list `acclist` for the bank account with the account name `accName`, and returns its balance. If `accName` is not in `acclist`, the method should throw a suitable exception.
Answer the Bank Account Question Here

(10 marks)
5.

Sudoku is a number placement puzzle: a 9 × 9 grid made up of 3 × 3 subgrids. Some cells already contain numbers, known as "givens", for example:

```
5 3 _ 7  
6 _ 1 9 5 
9 8 _ 6   
8 _ 6 _ 3 
4 _ 8 3 1 
7 _ 2 _ 6 
6 _ _ 4 2 8 
_ 4 1 9 5 
8 _ 7 9   
```

The goal is to fill in the empty cells, one number in each, so that each column, row, and subgrid contains the numbers 1 – 9 exactly once.

A player often needs to determine which numbers can be placed in a particular cell location. Given a partially filled game grid with 0s for empty cells, write a method

```java
public boolean[] findPossible(int[][] grid, int i, int j)
```

that returns a 9-element boolean array indicating whether each number is possible in the cell ith across and jth down from the top left corner (0,0). For example, if 1 is possible at the specified location, the first element of the boolean array should be true, otherwise false.

Factor out the subtasks into separate methods to make your code easier to understand.
Answer the Sudoku Question Here

(10 marks)
6.

(a) An array $a$ is said to dominate an array $b$ of the same length if every value in $a$ is at least as big as the corresponding value in $b$, and at least one value in $a$ is bigger than the corresponding value in $b$. For example, $\{1, 2, 3\}$ dominates $\{1, 0, 3\}$, but it does not dominate $\{1, 0, 4\}$.

Write a method

```java
public boolean dominates(int[] a, int[] b)
```

that returns true if $a$ dominates $b$, and false otherwise.
(b) The rank of an element $a[i]$ in a 2D array is equal to the number of elements of $a$ that dominate $a[i]$. (Note that multiple elements of $a$ can have the same rank.)

Write a method

```java
public int[] rank(int[][] a)
```

that returns an array containing the rank of each element of $a$.
7. Multiple Choice Questions [2 marks each]

(1) What is the value of the variable \( d \) after the following statements?

\[
\begin{align*}
\text{double } & \quad d = 0; \\
\text{int } & \quad i = 20; \\
\text{do } \{ \\
\quad & \quad d = d + (1.0/i); \\
\quad & \quad i = i/-2; \\
\} \text{ while } (i < 5); \\
\end{align*}
\]

A. This loop never terminates.
B. 0.0
C. 0.05
D. -0.05
E. 0.15

(2) In Java code that uses the term `java.Math.PI`, what does PI refer to?

A. It is a class variable belonging to the class `java.Math`.
B. It is a class method belonging to the class `java.Math`.
C. It is a method belonging to the object `java.Math`.
D. It is an instance variable belonging to the object `java.Math`.
E. It is the name of a class in the package `java.Math`.
(3) What is the value of the expression \( a == b \) after the following statements, and why?

```java
double[] a = new double[5];
double[] b = new double[5];

for (int i=0; i<5; i++) {
    a[i] = i*i;
    b[i] = i*i;
}
```

A. true because \( a \) and \( b \) are arrays of the same length with the same values.
B. true because they are both arrays of the same type.
C. false because they are both null since you can't assign integer values to variables of double type.
D. false because when applying == on double values, the possible loss of precision in floating point numbers may result in comparison errors.
E. false because \( a \) and \( b \) are different arrays, even though they have the same values.

(4) Which of the following methods returns true when exactly one of the three arguments is true, and returns false in all other cases?

A. `boolean exactlyOne(boolean a, boolean b, boolean c) {
    return (a || b) && (b || c) && (c || a);
}
`
B. `boolean exactlyOne(boolean a, boolean b, boolean c) {
    return !(a && b) || !(b && c) || !(c && a);
}
`
C. `boolean exactlyOne(boolean a, boolean b, boolean c) {
    return (a && !b && !c) || (!a && b && !c) || (!a && !b && c);
}
`
D. `boolean exactlyOne(boolean a, boolean b, boolean c) {
    return a || b || c;
}
`
E. `boolean exactlyOne(boolean a, boolean b, boolean c) {
    return (a || b || c) && !(a && b && c);
}
`
(5) What does int mystery(int a, int b) calculate, assuming a ≥ 0, b > 0?

    public int mystery(int a, int b) {
        if (b == 1) return a;
        else return a + mystery(a, b-1);
    }

A. It is a recursive way to calculate \(a - b\).
B. It is a recursive way to calculate \(a \times b\).
C. It is a recursive way to calculate \(a \% b\).
D. It is a recursive way to calculate \(a^b\).
E. It is a recursive way to calculate \(a + b\).

(6) Which of the following statements describe a program written in a good object-oriented style?

1. In carrying out their tasks, objects act as clients or servers but never both, during runtime.
2. Objects are capable of many complex tasks and communicate as little as possible during runtime.
3. Complex problems are solved by the co-operation of objects from several different classes.
4. Each class of objects provides a narrow range of well-defined services.

A. 1 and 2 only
B. 3 and 4 only
C. 2 and 3 only
D. 1 and 4 only
E. 2, 3 and 4 only
(7) The class `java.lang.Arrays` contains a method that checks if two arrays contain identical elements. The Java API describes this method as follows:

```java
public static boolean equals(int[] a, int[] a2)
    Returns true if the two specified arrays of ints are equal to one another. Two arrays are considered equal if both arrays contain the same number of elements, and all corresponding pairs of elements in the two arrays are equal. In other words, two arrays are equal if they contain the same elements in the same order. Also, two array references are considered equal if both are null.
```

Parameters:
- `a` - one array to be tested for equality.
- `a2` - the other array to be tested for equality.

Returns:
- `true` if the two arrays are equal.

Which of the following expressions uses this method to obtain the equality of two arrays `a` and `b`?

A. `a.equals(b)`
B. `Arrays.equals(a, b)`
C. `Integer.equals(a, b)`
D. `new int[a.length].equals(b)`
E. `Arrays.equals(new int[a.length], new int[b.length])`

(8) Which of the following is a true statement about constructors?

A. A class can have only one constructor.
B. A class can have zero explicitly specified constructors
C. A constructor is the same as a class method that returns nothing.
D. A constructor can return a primitive type.
E. A constructor cannot call an instance method.
(9) Consider these two expressions, where \( x \) is an \texttt{int}.

\[
(x \neq 5) \&\& (5 \div (x-5) > 10)
\]

\[
(5 \div (x-5) > 10) \&\& (x \neq 5)
\]

Which of the following is most correct about evaluating these expressions for different values of \( x \)?

A. Both expressions can always be evaluated successfully.
B. The second expression can always be evaluated, but for some values of \( x \), the first expression cannot be evaluated without causing a runtime error.
C. The first expression can always be evaluated, but for some values of \( x \), the second expression cannot be evaluated without causing a runtime error.
D. Neither expression can \textit{ever} be evaluated without causing a runtime error.
E. For a given value of \( x \), either both expressions can successfully be evaluated, or both will cause a runtime error.

(10) Suppose that we are attempting to sort the following array into \textit{increasing} order:

\[
\{10, 19, 2, 17, 3, 11, 7\}
\]

If we use \texttt{BUBBLESORT}, then what does the array contain after the first "pass"?

A. \(\{2, 3, 7, 10, 19, 17, 11\}\)
B. \(\{19, 17, 11, 10, 7, 3, 2\}\)
C. \(\{10, 2, 17, 3, 11, 7, 19\}\)
D. \(\{19, 7, 11, 3, 17, 2, 10\}\)
E. \(\{2, 3, 7, 10, 11, 17, 19\}\)

(11) When searching for an item in a sorted array containing 4,000 items, approximately how many comparisons must we make in the worst case if we use binary search?

A. About 13
B. About 8
C. About 22
D. About 256
E. About 1024
(12) Suppose the following method is incorporated into a class that draws on a SimpleCanvas (the variable c refers to a blank, newly-created 400 x 400 SimpleCanvas).

```java
public void drawPicture() {
    for (int i=0; i<400; i=i+20)
        c.drawLine(i,399,399,399-i);
}
```

Which of the following diagrams most closely resembles the picture that this method will draw?

A. 

B. 

C. 

D. 

E. 
(13) What is the value of method(8), where method is defined as follows?

```java
public int method(int n) {
    if (n == 1 || n == 2)
        return 1;

    int last = 1;
    int current = 1;
    int next;

    for (int i=3; i<=n; i++) {
        next = current + last;
        last = current;
        current = next;
    }

    return current;
}
```

A. 2
B. 8
C. 13
D. 21
E. 55

(14) Which one of the following lines of Java code may cause a compilation error?

A. short a = 14;
B. float a = 14;
C. String a = Hello;
D. boolean a = true;
E. int[][] a = {{1,2,3}, {4,5,6}};
Consider the following class, where each object represents an employee of a certain organization. The instance variable supervisor represents the supervisor of this employee (and is null for the CEO of the organization), while staff represents the number of staff who directly report to this employee. If this employee is an entry-level staff member, then he/she does not supervise anyone and has the value null for the variable staff.

```java
public class Employee {

    private Employee supervisor;
    private Employee[] staff;

    // irrelevant code omitted

    public int m() {
        if (this.staff == null) {
            return 1;
        } else {
            int total = 0;
            for (int i=0; i<staff.length; i++) {
                total = total + staff[i].m();
            }
            return total;
        }
    }
}
```

If Employee jan represents the staff member Jan, then what does jan.m() calculate?

A. The number of levels above entry-level that Jan has reached.
B. The number of staff subordinate to Jan (including Jan).
C. The number of staff who directly report to Jan.
D. The number of entry-level staff subordinate to Jan.
E. The number of staff subordinate to Jan (excluding Jan).
The next five multiple-choice questions refer to the following source code.

```java
/**
 * The Fraction class represents exact rational numbers such as 1/2, 2/5
 * by keeping them in fractional form. It supplies methods for
 * exact arithmetic using fractions.
 * 
 * Negative fractions are represented by storing the top as a negative
 * number; the bottom must always be strictly positive.
 * 
 * @author Gordon Royle
 */
public class Fraction
{
    private int top;
    private int bot;

    public final static Fraction ONE = new Fraction(1, 1);
    public final static Fraction ONE_HALF = new Fraction(1, 2);

    public Fraction(int top, int bot) {
        if (bot <= 0)
            throw new IllegalArgumentException("Zero or negative denominator");
        this.top = top;
        this.bot = bot;
        reduce();
    }

    public static Fraction add(Fraction alpha, Fraction beta) {
        int newTop = alpha.top*beta.bot + alpha.bot*beta.top;
        int newBot = alpha.bot*beta.bot;
        return new Fraction(newTop, newBot);
    }

    public static Fraction multiply(Fraction alpha, Fraction beta) {
        int newTop = alpha.top*beta.top;
        int newBot = alpha.bot*beta.bot;
        return new Fraction(newTop, newBot);
    }

    public Fraction reciprocal() {
        if (top == 0)
            throw new ArithmeticException("Division by zero");
        return (top < 0) ? new Fraction(-bot, -top) : new Fraction(bot, top);
    }
```
public double doubleValue() {
    return (double)top / (double)bot;
}

private void reduce() {
    if (top == 0)
        bot = 1;
    else {
        int g = (top < 0) ? gcd(-top, bot) : gcd(top, bot);
        top = top / g;
        bot = bot / g;
    }
}

private int gcd(int a, int b) {
    // see multiple choice question (18)
}

public String toString() {
    return top + "/" + bot;
}


(16) How many Fraction objects, in total, are created during the execution of the following sequence of statements?

Fraction a;
Fraction b;
a = new Fraction(11,14);
b = Fraction.add(a,a);
b = Fraction.multiply(a,b).reciprocal();

A. 1  
B. 2  
C. 3  
D. 4  
E. 5

(17) What is the result of executing the following statements?

Fraction a = new Fraction(1,-2);
Fraction b = Fraction.add(a,Fraction.ONE_HALF);
System.out.println(b.doubleValue());

A. The value 0.0 appears on the terminal window.  
B. Execution halts with an ArithmeticException.  
C. Execution halts with an IllegalArgumentException.  
D. Execution halts with an AssertionError.  
E. The value -0.5 appears on the terminal window.
At construction, a fraction is converted to its standard reduced form by working out the greatest common divisors of the denominator and the numerator. Which of the methods below will correctly return the greatest common divisor for two positive integers?

//Method 1
private int gcd(int a, int b) {
    if (a < b) return gcd(b, a);
    if (b == 0) return a;
    return gcd(b, a % b);
}

//Method 2
private int gcd(int a, int b) {
    if (a < b) return gcd(b, a);
    if (b == 0) return a;
    return gcd(b, a / b);
}

//Method 3
public static int gcd(int a, int b) {
    if (a > b) return gcd(b, a);

    int gcd = 1;
    for(int i=1; i<=a; i++)
        if(b%i == 0 && a%i == 0) gcd = i;
    return gcd;
}

A. Method 1 only.
B. Method 2 only.
C. Method 3 only.
D. Methods 1 and 3.
E. Methods 2 and 3.
(19) What are the values of a.top and a.bot respectively, after the following statements?

```java
Fraction a = new Fraction(-12,16);
a = Fraction.multiply(a,Fraction.ONE_HALF).reciprocal();
```

A. -8 and 3  
B. 3 and -8  
C. 8 and -3  
D. -32 and 12  
E. 32 and -12  

(20) Consider the following method.

```java
public boolean mystery(Fraction alpha){
    return (alpha.top*bot - alpha.bot*top) > 0;
}
```

What does the mystery method do? What will be printed out on the terminal after the following sequence of statements?

```java
Fraction a = new Fraction(12,16);
Fraction b = new Fraction(1,4);
System.out.println(a.mystery(b));
```

A. It returns true if the numerical value of the argument fraction alpha is greater than this fraction. false is printed on the terminal.  
B. It returns true if the numerical value of this fraction is greater than the argument fraction alpha. true is printed on the terminal.  
C. It returns true if the numerical value of the argument fraction alpha is greater than this fraction. true is printed on the terminal.  
D. It returns true if the numerical value of this fraction is greater than the argument fraction alpha. true is printed on the terminal.  
E. It compares the numerical value of this fraction and the argument fraction. true is printed on the terminal.
BLANK PAGE FOR ROUGH WORK
BLANK PAGE FOR ROUGH WORK
BLANK PAGE FOR ROUGH WORK