

Candidate number: _____

ENGLISH

UNIVERSITY OF OSLO

Faculty of mathematics and natural sciences

Exam in : INF1000 - Introduction to object-oriented programming
Day of exam : Monday, December 3, 2007
Time : 09 – 12
Number of pages : 12 pages
Appendices : None
Allowed to bring with you : All written and printed material

- Read *carefully* through each question before you try to answer it. Each question is marked with a maximal score given for an entirely correct solution. The total max score is 252 points, which means that 4 points correspond to approximately 3 minutes and 10 points correspond to approximately 7 min (should you plan to get through all the questions). Try to use your time well!
- Make sure that your examination set is complete before you answer any questions. If there is information missing in the formulation of an exercise, you may make your own assumptions as long as they are in accordance with the “spirit” of the exam. Such assumptions must be explained.
- You shall write your answers directly onto these pages, not on separate sheets of paper. This applies to all questions, both multiple choice questions and questions where you are asked to write program code. In the questions where you are asked to write program code, we recommend that you first make a draft on a separate sheet of paper and then write your answer on this examination form in the appropriate place.
- Some of the questions are multiple-choice questions. For these questions your score depends on how many correct answers you give. No score is given if you leave a question unanswered, or if you mark both alternatives.
- If you have marked an alternative with a cross and you later determine that you do not want a cross there, you can write "ERROR" immediately to the left of the box.
- You must write hard enough to make all copies that you hand in readable. There must be no other papers from the examination text underneath when you write!

Question 1 (2 points)

How many int-values can be stored in the array 'horses'?

```
int[][] horses = new int[10][5];
```

Answer:

Question 2 (20 points)

Are these program statements allowed in Java?

- | YES | NO |
|--------------------------|--|
| <input type="checkbox"/> | <input type="checkbox"/> <code>float x = 3 / 2;</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>int x = 3 / 2;</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>int[]boolean[] s = new int[8]boolean[9];</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>int i =1, j=i+3, while(j-->0) i = j;</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>long[] boolean = new boolean[12];</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>boolean[] x = new boolean[0];</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>int XX = (int) (3.1415);</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>double x = (int) 33.5;</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>String[3] a = {"1", "2", "3"};</code> |
| <input type="checkbox"/> | <input type="checkbox"/> <code>int[][] a = new int[][10];</code> |

Question 3 (16 points)

a) How many times is “INF1000” printed out in the following loop:

```
for (int i=1; i<6; i++) {  
    System.out.println("INF1000");  
}
```

Answer: times

b) How many times is “INF1000” printed out in the following loop:

```
int k = 1;  
while (++k < 10)  
    System.out.println("INF1000 ");
```

Answer: times

c) How many times is “INF1000” printed out in the following loops:

```
for (int j=1; j < 10 ; j = j+1 ) {  
    for (int i = j; ++i < j+1; j = i) {  
        System.out.println("INF1000");  
    }  
}
```

Answer: times

d) How many times is “INF1000” printed out in the following loop:

```
int k = 1;  
while (++k +1 < 9 )  
    System.out.println("INF1000");
```

Answer: times

Question 4 (4 points)

Assume that the following lines of code are executed:

```
int one = 1, two=2, three=3;
int four = one++ + ++two - --three;
int answer = four + 3*three + 2*two + 1*one;
```

What is the value of the variable **answer** right afterwards?

Answer:

Question 5 (20 points)

Finish the method below, which calculates and returns the volume V of a sphere from the radius of the sphere and π . The formula you will use is: $V = \frac{4}{3}\pi r^3$, where $\pi = 3.1415926$ and r is the radius of the sphere. (*Hint*: Make sure that you do not mix up integer and floating point division.)

Svar:

```
double sphereVolume( double radius ){
}
}
```

Question 6 (20 points)

Finish the method below, which starting from an array with the prices for a barrel of oil for some days of a month, calculates the average price for that month and returns this average price. The problem is that for some of the days, the prices are missing and the number -1.0 is in that position in the array instead of a real price (which we assume is a positive number). The average price is defined as the sum of all the real prices divided by the number of such prices in the array. The prices in the array are given as floating point numbers and in US dollars. Your answer for the average price should be in Norwegian Kroner. You will use the currency rate for the dollar of NOK 5,28 (i.e. 1 US dollar is worth NOK 5,28).

Question 8 (15 points)

```
int rest(int n, int m){
    int k = 0;
    while(n >= m) {
        n-=m;
    }
    while ( n-- > 0) {
        k++;
    }
    return k;
}
```

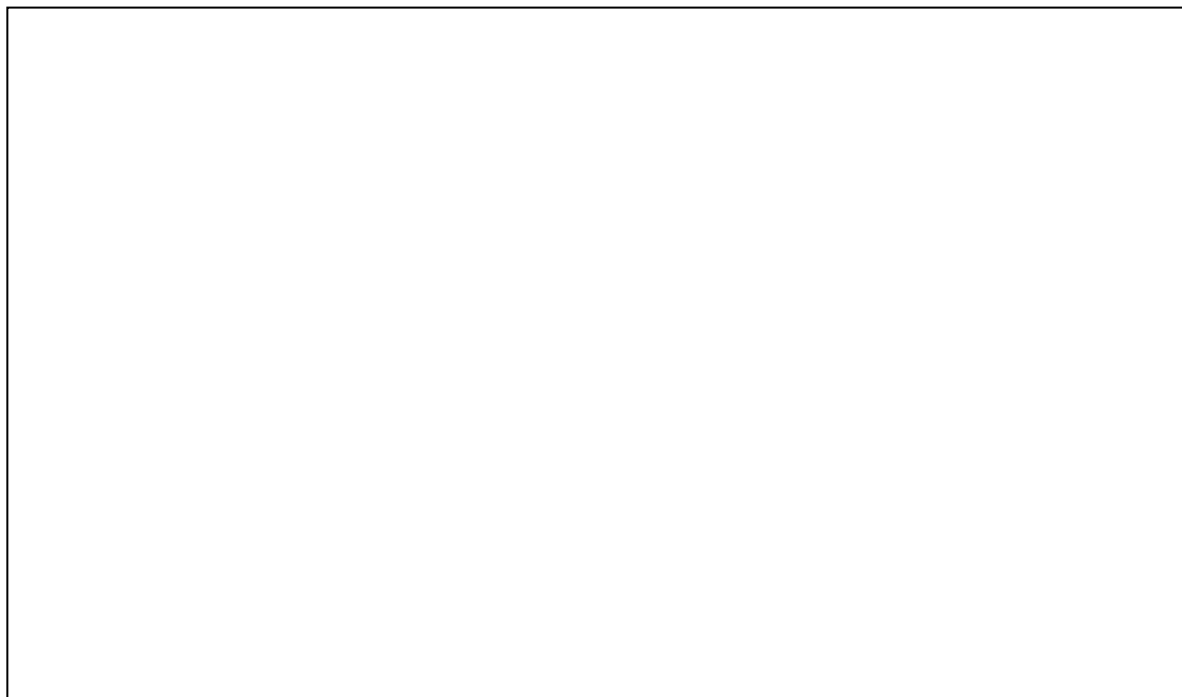
What is returned from the method call `rest(41,7)`?

Answer:

Question 9 (20 points)

We are getting close to Christmas and Santa is very busy. He has also completely lost control of his directory of all the children in the world and their wishes. This is something that he will need when Christmas night comes. Santa has given the elves the assignment of creating a new computer system for him to organize the information that he needs on Christmas night. Your assignment is to help the elves draw a UML class diagram for Santa's computer system. The whole world consists of 5 continents. Santa does not manage to visit all the continents himself (Sorry if we spoil some of the Christmas joy for you!) so on Christmas night there are 10-100 elves covering each continent for Santa. On every continent there lives none or more good children. A good child does not have to have a wish, but Santa will grant no more than 5 wishes. There is no room for any bad children in Santa's computer system. Draw a UML class diagram with the 5 (Java) classes that will be used to represent this problem. Name both the classes and the relations between them and place numbers on both sides of the relation between the classes.

Answer:



Question 10 (60 point)

All the students at the University of Tlön in the country Uqbar will take from 1 to 5 exams each semester, and since all the students have unique login names on the university's computer system, all the exam results (A-E and F) of a student is registered with this user name (i.e. the student Arno Mår has the user name 'arnom'). All the results for one exam is registered in a file (also those that get an F). The name of this file is the name of the course followed by '.res' (i.e. the exam results for INF1000 is stored on the file "**INF1000.res**" and for MAT1000 in the file: "**MAT1000.res**", and so on). On a file like that there is a line per student which took the exam in the following format:

```
<username> <grade>
```

For example: If Arno Mår got an E in INF1000 and failed in MAT1000, there would be a line in the file: "**INF1000.res**" that reads:

```
arnom E
```

and a line in the file: "**MAT1000.res**" that reads:

```
arnom F
```

The names of all the exam files for the current semester is on a file: "**AllTheExams.txt**" with a line for each file name (i.e. **MAT1000.res**).

The system that you will write will be used to send *one single e-mail* to each of the students that have taken an exam this semester with *all* the exam results for that student this semester. This means that Arno Mår will receive an e-mail titled "Your 2 exam results this semester" and the text of the e-mail will be: "Hello arno, you got an E in INF1000, and you got an F i MAT1000. Kind regards, the exam system at the Universitetet of Tlön".

To help you send e-mails someone has written the package 'easyEmail', which among other things contain a class **Email** which has a method (which is not static):

```
boolean sendAnEmail(String username,String title,String message)
```

which obviously sends an e-mail titled: **title** and with content/message: **message** to: **username** (the method appends the local suffix for the address so that it becomes a complete and valid e-mail address.: username@ifi.uio.no). The return value from the method tells you whether the e-mail system found this user or if he or she does not exist. Your system must write an error message to the screen ever time a wrong/non-existing username is found.

We demand from your solution that it has at least two classes: **ExamSystem** and **SendEmails** (but probably more). Your program will first go through a loop and for each exam file in "**AllTheExams.txt**" it will call a method that reads that file and which for each student in the exam file, manages to store in a convenient way the *exam result* and *which exam* and *which student* this concerns. After the method has been called for all the exams, another method will be called which for each student registered with an exam this semester, first builds the message to the person and then calls the **sendAnEmail**-method in the class **Email** to send this message.

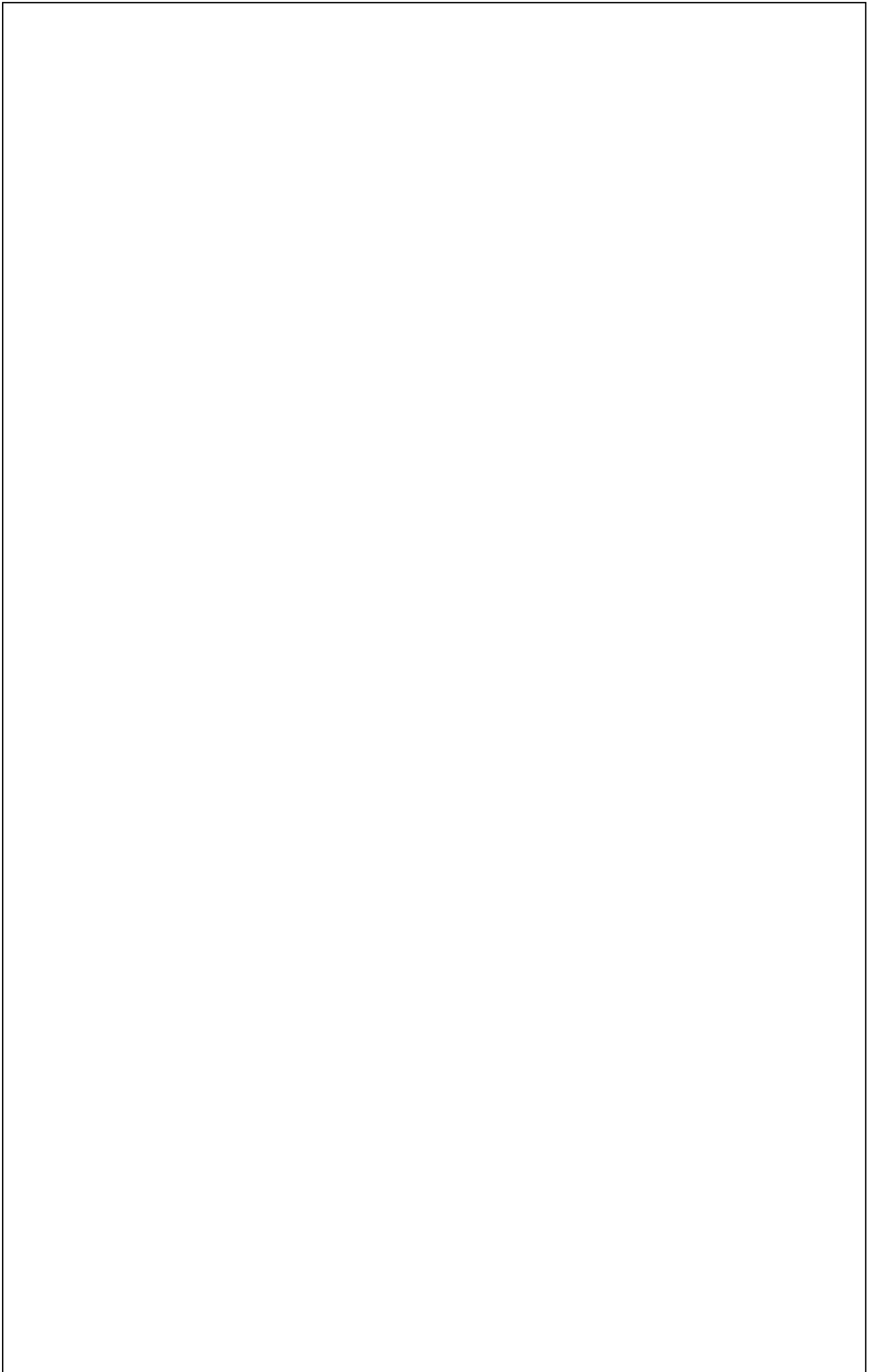
Answer:

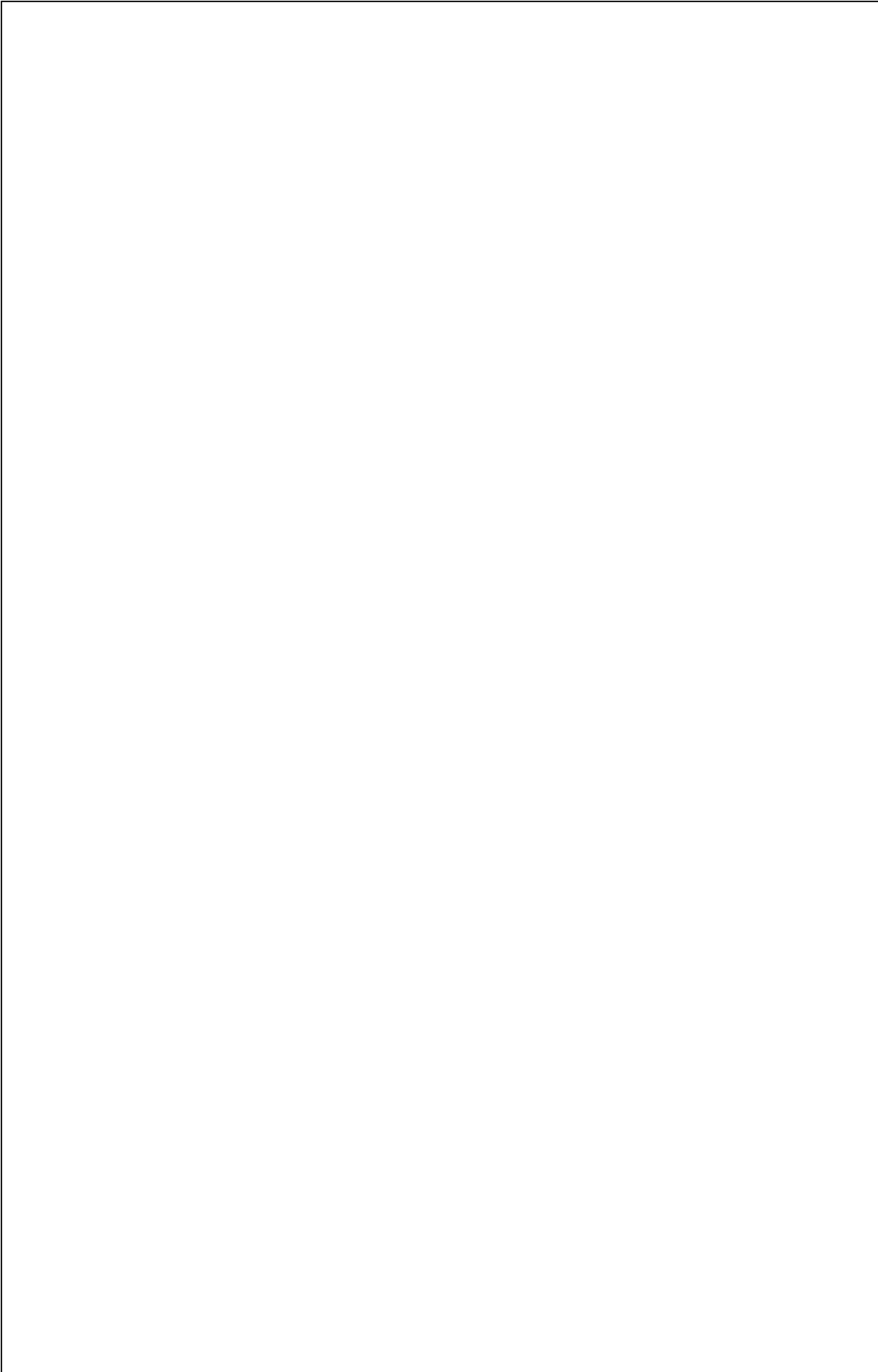
```
import easyIO.*;
import easyEmail.*;
import java.util.*;

class ExamSystem {
    public static void main(String [] args) {
        SendEmails s = new SendEmails("AllTheExams.txt ");
        s.sendMessage();
    }
} // end ExamSystem

class SendEmails {

    SendEmails (String examFile) {
```





Question 11 (15 points) Slightly difficult

Assume that the following program is executed:

```
class Tree {
    Tree left;
    Tree right;
    String text;
    Tree(String text){
        this.text = text;
    }
    Tree(String text, Tree left, Tree right){
        this.text = text;
        this.left = left;
        this.right = right;
    }
    void print(){
        if(this.left != null)
            this.left.print();
        System.out.print(" " + this.text + " ");
        if(this.right != null)
            this.right.print();
    }
    public static void main(String[] args) {
        Tree a = new Tree("A");
        Tree b = new Tree("B");
        Tree c = new Tree("C", a, b);

        Tree d = new Tree("D");
        Tree e = new Tree ("E", c, d);

        c.print();

        System.out.print(" *** ");

        e.print();
    }
}
```

First, make a drawing for your self and find out how many objects of the class **Tree** that are made by the sentences in main and what the values of the object variables of these objects are. Then answer the question: what does the program print to the screen?

Answer:

Question 12 (30 points)

You will write two methods **findMinIndex** and **sort** which together sorts an array **a** in the following way: First, we find the index in the array **a** where the smallest element of **a** is. Then we let that element and the element in **a[0]** switch places. Then we find the second least element of the rest of **a** – which are the ones in the positions 1,2,..., and we let that element switch places with the element in **a[1]**,... and so on until the entire array **a** is sorted. The user will thus get a sorted array by calling the method **sort**.

N.B. You will receive no points for copying the sorting method from the book.

Svar:

```
int findMinIndex (int[] a, int start, int end) {
```

```
}
```

```
void sort (int[] a){
```

```
}
```

Question 13 (20 points)

The country Uqbar has exactly the same “Law about treating personal information” (“Lov om behandling av personopplysninger”) as we do in Norway and you will use this law to address the following:

The exam system at the University of Tlön in Uqbar will now use the data that are in the files in question 10 to make a system that reports to the State Educational Loan Fund (Lånekassa) in Uqbar the students that do not have a good enough progress in their education by **not** passing at least 3 or more exams this semester. They will for each such student send an email to the State Educational Loan Fund (Lånekassa) about how many exams these students have passed. You will **not** write the code for this, but rather write an evaluation about whether the University of Tlön in Uqbar is allowed, from the legislation in the “Lov om behandling av personopplysninger”, to send such messages about students that do not work hard enough to Lånekassa, which will result in the students getting smaller loans next semester. Defend your position by referring to specific paragraphs in the law that you think are relevant and why they are relevant: