University of London International Programmes
Computing coursework assignments 2015–16

CO3352 Operations research and combinatorial optimisation
Coursework assignment 1

This coursework assignment is designed to help you enrich your learning experience and to encourage self-study and creativity. Assumptions may be added if necessary in your coursework answers to simplify programming tasks or help with understanding issues. You should, however, attempt the exercises in the subject guide before doing the coursework. Otherwise, you may find the tasks in the coursework assignment difficult and the experience less rewarding. You should read the coursework assignments or questions carefully and pay particular attention to the Submission Requirements on page 3. Note: programme source code may be checked using plagiarism-detecting programs. The available marks are given in square brackets.

1. Provide an example of each of the listed matroids below. Let the ground set be $E = (a, b, c, d)$. List the members of the independent set $\mathcal{I}$ for each of your examples. Draw a diagram if your example is based on a graph. [40%]

   (a) A cycle matroid
   (b) A graphic matroid
   (c) A cographic matroid
   (d) A transversal matroid
   (e) A matroid that is *not* representable
   (f) A partition matroid
   (g) A uniform matroid
   (h) A matching matroid

2. Develop a prototype program in Java (or MATLAB) to demonstrate how Algorithm 121 works (see page 46 of the CO3352 Subject guide).
   There is no specific requirement for your user interface design. For example, no input-error-handling is required. However, your program should prompt the user to input m and n, and display some relevant intermediate values such as X, C, Z for information. [60%]

[END OF COURSEWORK ASSIGNMENT 1]

University of London International Programmes
Computing coursework assignments 2015–16

CO3352 Operations research and combinatorial optimisation
Coursework assignment 2

This coursework assignment is designed to help you enrich your learning experience and to encourage self-study and creativity. Assumptions may be added if necessary in your coursework answers to simplify programming tasks or help with understanding issues. You should, however, attempt the exercises in the subject guide before doing the coursework. Otherwise, you may find the tasks in the coursework assignment difficult and the experience less rewarding. You should read the coursework assignments or questions carefully and pay particular attention to the Submission Requirements on page 3. Note: programme source code may be checked using plagiarism-detecting programs. The available marks are given in square brackets.

TreeSurgeon is a logging company that will soon begin logging eight groves of trees on a new site. The company therefore needs to develop a system of dirt roads to make each grove accessible from every other grove. Let the groves be labelled as $(1, 2, \cdots, 8)$. The distance (in miles) between every pair of groves is shown in the table below. For example, the distance from grove "3" to grove "5" is 1.7 miles as highlighted in a box in the table.

Grove	1	2	3	4	5	6	7	8
1	0	1.3	2.1	0.9	0.7	1.8	2	1.5
2	1.3	0	0.9	1.8	1.2	2.6	2.3	1.1
3	2.1	0.9	0	2.6	1.7	2.5	1.9	1.0
4	0.9	1.8	2.6	0	0.7	1.6	1.5	0.9
5	0.7	1.2	1.7	0.7	0	0.9	1.1	0.8
6	1.8	2.6	2.5	1.6	0.9	0	0.6	1
7	2	2.3	1.9	1.5	1.1	0.6	0	0.5
8	1.5	1.1	1	0.9	0.8	1	0.5	0

The task is to determine between which pairs of groves the roads should be constructed to connect all groves with a minimum total length of the roads.

1. Formulate an OR problem for the instance above and write a problem statement for TreeSurgeon. [10%]
2. Outline a feasible algorithmic solution to the OR problem described in part 1. [30%]
3. (Programming task) Develop a program in Java (or MATLAB) to fulfil the following subtasks: [60%]

   (a) Display your graph model in terms of the adjacency matrix.
   (b) Display the solution in terms of the adjacency matrix, or a set of vertex pairs.

[END OF COURSEWORK ASSIGNMENT 2]

Submission requirements

These requirements apply to both coursework assignments. The available marks are given in square brackets.

1. Naming conventions for any `.pdf` or `.zip` file submissions
   When naming your files, please ensure that you include your full name, student number, course code and assignment number, e.g. `FamilyName_SRN_COxxxxcw#.pdf`
   (e.g. `Zuckerberg_920000000_CO3352cw2.pdf`), where

   - `FamilyName` is your family name (also known as last name or surname) as it appears in your student record (check your student portal)
   - `SRN` is your Student Reference Number, for example 920000000
   - `COXXXX` is the course number, for example CO3352, and cw# is either cw1 (coursework 1) or cw2 (coursework 2).

2. Your coursework submission must include a report Document [20%] and the program Code [40%].
   The Document (preferable in .pdf format) should include the following sections:

   (a) Algorithms (in flow-chart)
   (b) Design (in block diagram or class-diagram in UML)
   (c) Demonstration (in 5 best screen-shots)
   (d) Discussion (including answers to any questions/problems in the Coursework assignment, your experience in attempt of the coursework, and full bibliography)

   The program code should include the

   (a) Java source codes .java
   (b) executable version .class.

3. Execution of your programs:
   [Penalty] A ZERO mark may be awarded if

   - your program(s) cannot be run from the coursework directory by a simple command
     '*java menu*' (this means that you should name your main class 'menu', or adopt the `menu.java` that can be found in the Appendix on page 4);
   - your source code(s) does not compile and you give no information on your program execution environment;
   - your program(s) does not do what you claim it should do;
   - your program(s) crashes within the first *three* interactive execution steps;
   - your program(s) works for the first time of execution only;
   - there is no comment in your source code.

4. You should monitor and report the time you have spent for each part of the coursework answers, and leave a note to the examiner if you need to raise any issue at the beginning of your coursework answers as follows:

Total Number of Hours Spent	
Hours Spent for Algorithm Design	
Hours Spent for Programming	
Hours Spent for Writing Report	
Hours Spent for Testing	
Note for the examiner (if any):	

5. Show *all* your work. Any use of others' work should be declared at the point of use and referred to in the *Bibliography* section at the end of your coursework answers.

Appendix

This is an example. Please modify accordingly to suit your own purposes.

```java
import java.lang.*;
import java.io.*;
// Modify the display content to suit your purposes...
class menu {
private static final String TITLE =
"\nCO3352 coursework\n"+
"    by firstname-FAMILYNAME_SRN\n\n"+
"\t********************\n"+
"\t1. Declaration: Sorry but part of the program was copied
from the Internet! \n" +
"\t2. Question 2 \n"+
"\t3. Question 3 \n"+
"\t4. no attempt \n"+
"\t0. Exit \n"+
"\t********************\n"+
"Please input a single digit (0-4):\n";
menu() {
int selected=-1;
while (selected!=0) {
System.out.println(TITLE);
BufferedReader in = new BufferedReader(new InputStreamReader(System.in));
// selected = Integer.parseInt(in.readLine());
try {
                    selected = Integer.parseInt(in.readLine());

                        switch(selected) {
                            case 1:  q1();
                              break;
                            case 2:  q2();
                              break;
                            case 3:  q3();
                              break;
                            case 4:  q4();
                              break;}        }
   catch(Exception ex) {}  } // end while
            System.out.println("Bye!");
}
// Modify the types of the methods to suit your purposes...
private void q1() {
      System.out.println("in q1");
}
private void q2() {
System.out.println("in q2");
}
private int q3() {
System.out.println("in q3");
return 1;
}
private boolean q4() {
System.out.println("in q4");
return true;
}
   public static void main(String[] args) {
new menu();
   }
}
```

[END OF SUBMISSION REQUIREMENTS]