Computational Thinking and Program Design
Individual Class Project
(Due at 23:59 on 14 June 2022)

Rocky K. C. Chang

17 April 2022

Objective:

You will gain first-hand experience on the entire problem-solving process through designing and implementing a program to solve a river-crossing puzzle.

The Project:

In this project you will design and implement a program to solve a river-crossing puzzle similar to the MCGW problem with slightly higher complexity. There are six entities: Man, ape, bull, rhino, goat and cabbage. The puzzle is to bring all of them from the east side of a river to the west in a smallest number of boat trips. Several constraints below:

1. The boat can take at most two entities.
2. Only the man and the ape can row the boat.
3. The ape cannot stay alone with the bull without the man.
4. The ape cannot stay alone with the goat without the man.
5. The bull cannot stay alone with the rhino without the man.
6. The goal cannot stay alone with the cabbage without the man.

You must apply and document the four-step process: data abstraction, algorithm design, program design and implementation. In the program design, you must modularize your code by identifying the functions that you need and their signatures. Your program must print only one of the solutions in the exact wording and format as below.

```
1.  The ape takes the cabbage from the east to the west.
2.  The ape takes only itself from the west to the east.
3.  The ape takes the rhino from the east to the west.
4.  The ape takes only itself from the west to the east.
5.  The man takes the ape from the east to the west.
6.  The man takes the cabbage from the west to the east.
7.  The man takes the goat from the east to the west.
8.  The man takes the ape from the west to the east.
9.  The man takes the bull from the east to the west.
10. The man takes the rhino from the west to the east.
11. The man takes the ape from the east to the west.
12. The ape takes only itself from the west to the east.
13. The ape takes the cabbage from the east to the west.
14. The ape takes only itself from the west to the east.
15. The ape takes the rhino from the east to the west.
```

Deliverables:
1. A report documenting the process of solving the problem
   a. Three pages maximum + one page for the graph. One inch on all margins and Times New Roman font with 11pt and single space.
   b. The report must contain the following information and sections. There is no need to include the code, because it is already in the *.py* file.
      i. Your name, student ID and program name
      ii. A problem description
      iii. Data abstraction (including a description of the states, a graph as a result of the data abstraction, the data types required, …)
      iv. The algorithm needed to solve the graph problem
      v. A modular design of the program
      vi. A Python implementation of the data types
2. A well documented Python program in a single *.py* file for the main program and possibly another *.py* file for a library
   a. You must use docstring to describe each function.
   b. By using appropriate comments and variable names, your program must be easy to follow and understand.
   c. Give proper reference to the source of the code that you adopt for your program.

Rubric for grading:

	Expectations fully met (4)	**Expectations mostly met (3)**	**Expectations somewhat met (2)**	**Expectations largely not met (1)**	**Expectations not met at all (0)**
Program Correctness (30%)	The program runs correctly according to the specification for all cases.	The program runs correctly according to the specification almost for all cases.	The program runs correctly according to the specification only for the major cases.	The program runs correctly according to the specification only for a small number of cases.	The program does not run correctly for all cases.
Code readability and documentation (30%)	The code is very well organized and very easy to understand. The documentation is very clear and accurate.	The code is generally well organized and easy to understand. The documentation is generally clear and accurate.	The code is adequately organized and needs some effort to understand. The documentation is adequately prepared.	The code is poorly organized and is difficult to understand. The documentation lends very little help.	No effort to organize the code for understanding. There is no/very little documentation.
Content of the report (40%)	The report documents all the main steps of problem solving very clearly, logically and comprehensively.	The report documents most of the main steps of problem solving clearly, logically and comprehensively.	The report meets just the basic requirements in terms of clarity, logical presentation and comprehensiveness.	Lacking important details and clarity, the report offers limited help in understanding the problem-solving process.	The report is largely incomprehensible. It offers no help in understanding the problem-solving process.