

Sultan Qaboos University
College of Science - Department of Computer Science
COMP2101: Introduction to Computer Science Spring 2006
Programming Assignment 4
Due date: 05 April 2006, 23:55 pm

Important instructions:

- 1) Time allocated for this assignment is two weeks.
- 2) Each student must submit his/her own solution of the programming assignment.
- 3) Students who submit the same assignment will receive a score of 0.

SUGGESTED LAB EXERCISES: Solve Exercises: **3.11, 3.22, 3.28, 3.30a, 3.30b**

Problem:

The number of direct roads needed to maintain a directly connected N number of cities, is given by the formula:

$$\text{Roads} = N \times (N - 1) \div 2$$

Write a complete C++ program that reads a set of **unspecified** number of N values stored in file (**cities.txt**), and calculates the number of direct roads needed for each N . The result is then printed to both, screen as well as to the output file (**roads.txt**).

Your program **MUST** use at least the following **files** and **functions** with suitable parameters:

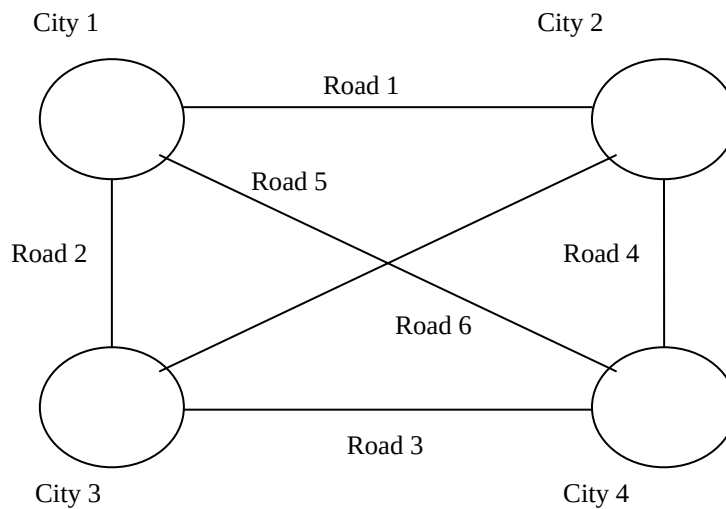
- **cities.txt**: input file that stores different values of N (positive integers) with each N representing the number of cities.
- **roads.txt**: output file to save the resulting numbers of direct roads needed for each N (using the format shown in the sample output below).
- **GetCities()**: a function that reads and passes to main (), individual values of N from file **cities.txt**.
- **ComputeRoads()**: a function that uses the above formula to compute and return the required number of direct roads for each value of N .
- **PrintToFile()**: a function that prints the result to file **roads.txt**.
- **PrintToScreen()**: a function that prints the result to the standard output stream.

Your program **MUST** also do the following:

- Check and ignore invalid values of N .
- Use call-by-value and call-by-reference at least once when using functions.
- Output the result using the format shown in the sample output below.

Example:

Directly connecting 4 cities ($N = 4$) requires 6 individual roads as shown in the figure below ($Roads = 4 \times (4-1) \div 2 = 6$):

**Sample Input and Output**

Input (file cities.txt)	Output (file roads.txt) && Screen	
2	Welcome to the Road System	
-1	-----	
4		
6	Number of Cities	Number of Direct Roads
	-----	-----
	2	1
	4	6
	6	15
	Thank you for using our system ...	

Grading Table

	0	0.25	0.5	0.75	1
<i>Style (comments, naming, indentations...)</i>					
<i>Declarations (including files)</i>					
<i>Functions prototypes</i>					
<i>GetCities() definition</i>					
<i>ComputeRoads() definition</i>					
<i>Check for invalid input and errors in files)</i>					
<i>Using call-by-value</i>					
<i>Using call-by-reference</i>					
<i>PrintToFile() definition</i>					
<i>PrintToScreen() definition</i>					