PROJECT 4 EVALUATION RUBRIC

Program	Description	Points
Documentation	Main Comment Block contains: (author (1) and program description (1)).	2
	Comments have been added to each group of logically related statements **Comments are written above each user defined function to describe what the function does**	3
Style	Variable: • Meaningful variable names are used unless specified by the program description (1) • No global variable is used (1) Function • Meaningful function names are used (1) • In the client program, function prototypes declared above the main function and function definitions written after the main function (2)	5
	Indentation and white spaces are used to make the program easier to read. • All the decision statements are indented properly. • All the repetition statements (loops) are indented properly • Body of the functions are indented properly • Blank lines are used in front of each block of logically related statements	4
	Array and vector size should be declared as a constant.	1
Correctness	Program solves the assigned problem using data structure and methods described in project description.	35
	Program compiles without errors.	5
	Program executes without crashing.	5
	Program produces the correct output in table format as shown in the example program output.	40
TOTAL		100

Program requirements:

1. Use STL stack class in the development of the algorithm for deriving the itinerary
2. In the flight map ADT (FlightMapClass) created in Project 3. Add the following functions and data:
   - the non-recursive IsPath algorithm discussed in class to find the itinerary between two cities. Modify the code to display the full itinerary if one is found.

- Additional functions used by the IsPath algorithm need to be added to the FlightMapClass (e.g., MarkVisited, IsVisited, UnvisitAll, GetNextCity…) see the updated FlightMapClass.h.
- Additional data for the class: visited array to record whether a city has been visited during the itinerary planning process.