### Chapter 3: Testing Results

Here I design 3 special cases and many random input cases to test the algorithm, the specification is shown in the table below. During the testing stage, some input has given the output contradict to the expected outcome. After debugging and analysis, we corrected the problems.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| case | input | output | expected | description | status |
| 1 | N=1000,M=499500  2,1,1,1  3,1,1,1  3,2,1,1  ……….  1000,999,1,1 | 0 | 0 | I design the case to test if the algorithm can handle the worst case that all cities are connected to each other with a road in use of cost 1,however,the country is still connected after a city is destroyed. | Pass |
| 2 | N=502,M=1001  2,1,1,1  3,1,1,1  4,1,1,1  ……  3,2,1,0  4,2,1,0  …… | 1 | 1 | Each city except 1 is connected to 1 with a road of cost 1. City 3 to 502 has a destroyed road of cost 1 to 2. This case can test if the algorithm can find the aim correctly in a lot of cities and find the cost to reconnect the country for each city destroyed. | Pass |
| 3 | N=8,M=11  1,2,1,1  2,3,1,1  3,4,1,1  4,5,1,1  5,6,1,1  6,7,1,1  7,8,1,1  1,3,1,0  3,5,1,0  5,7,1,0  6,8,1,0 | 2,4,6,8 | 2,4,6,8 | This case is designed to test if the algorithm can output several cities together | Pass |
| Random  conditions | …… | ……. | …… | To test the accuracy of the algorithm | All pass |

### Chapter 4: Analysis and Comments

The time complexity of the algorithm is O(MN),for a prim algorithm is used to generate the minimal spanning tree if a city is destroyed during the war. Each generation will cost O(M) and N generation is needed for comparison.

The space complexity is relatively large ,because we use a lot of arrays in our algorithm for simplicity. For example, the array used to store the edges between cities can be as large as O(N^2).

Comments:

It is obvious that the space complexity can be reduced dramatically by using a adjacency list instead of array. Thus we can reduce the complexity to O(M). Afterall ,the case here is too horrible. And the the array will also cause a waste of time in initialization .

Finding minimum tree and minimum cost functions can be improved by using a heap.