COCSC06(DAA)-LAB FILE

netaji subas university of technology | NEW DELHI

By: amogh garg

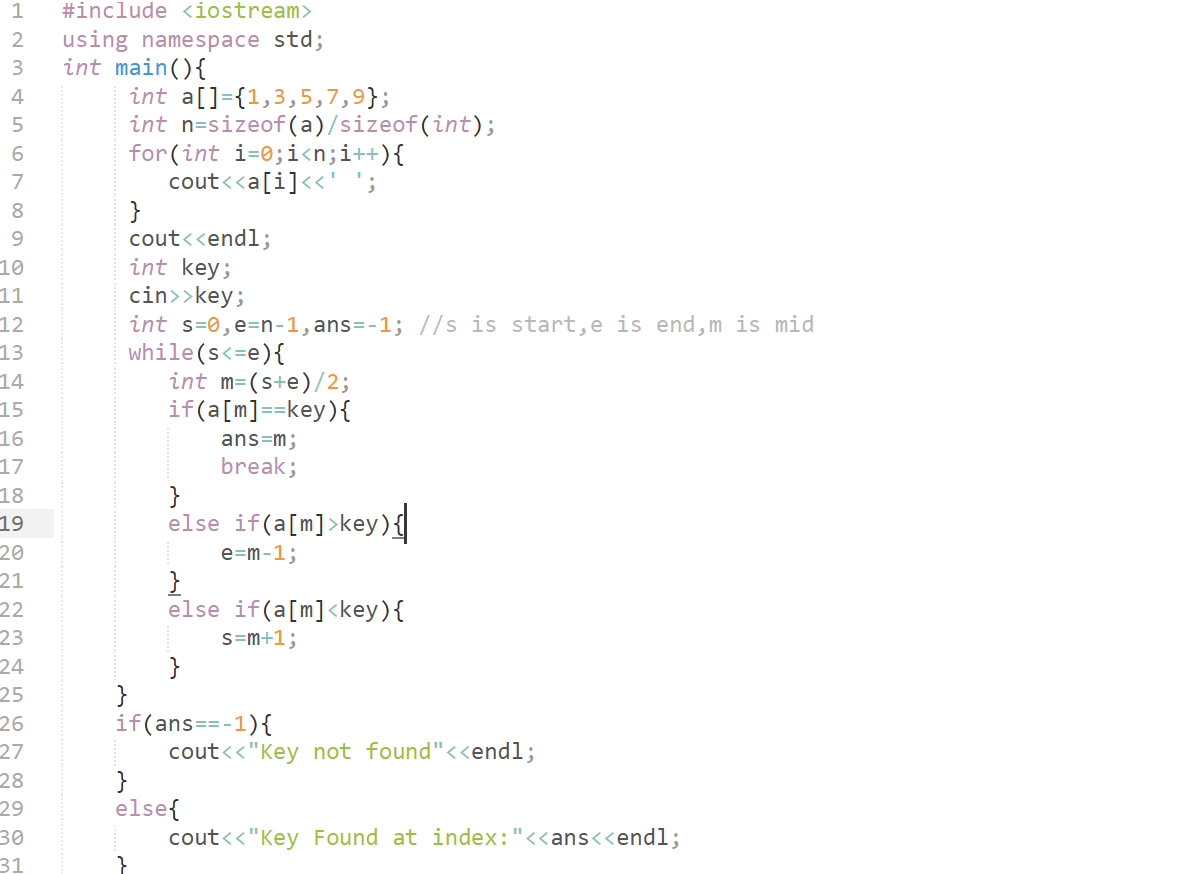
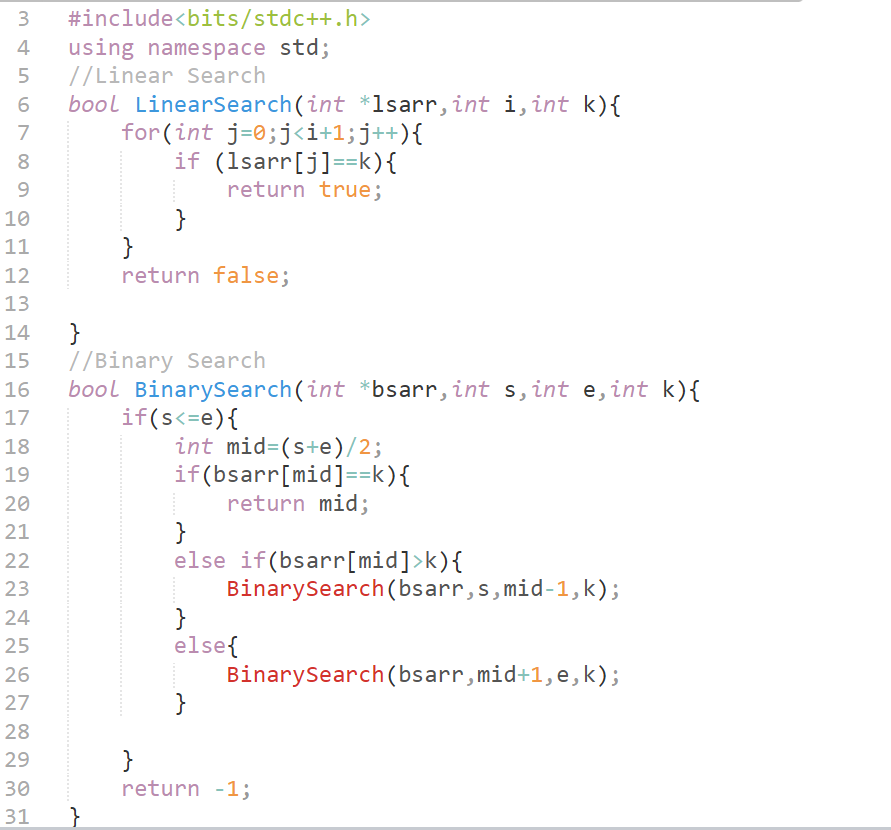
Roll number: 2020uco1688

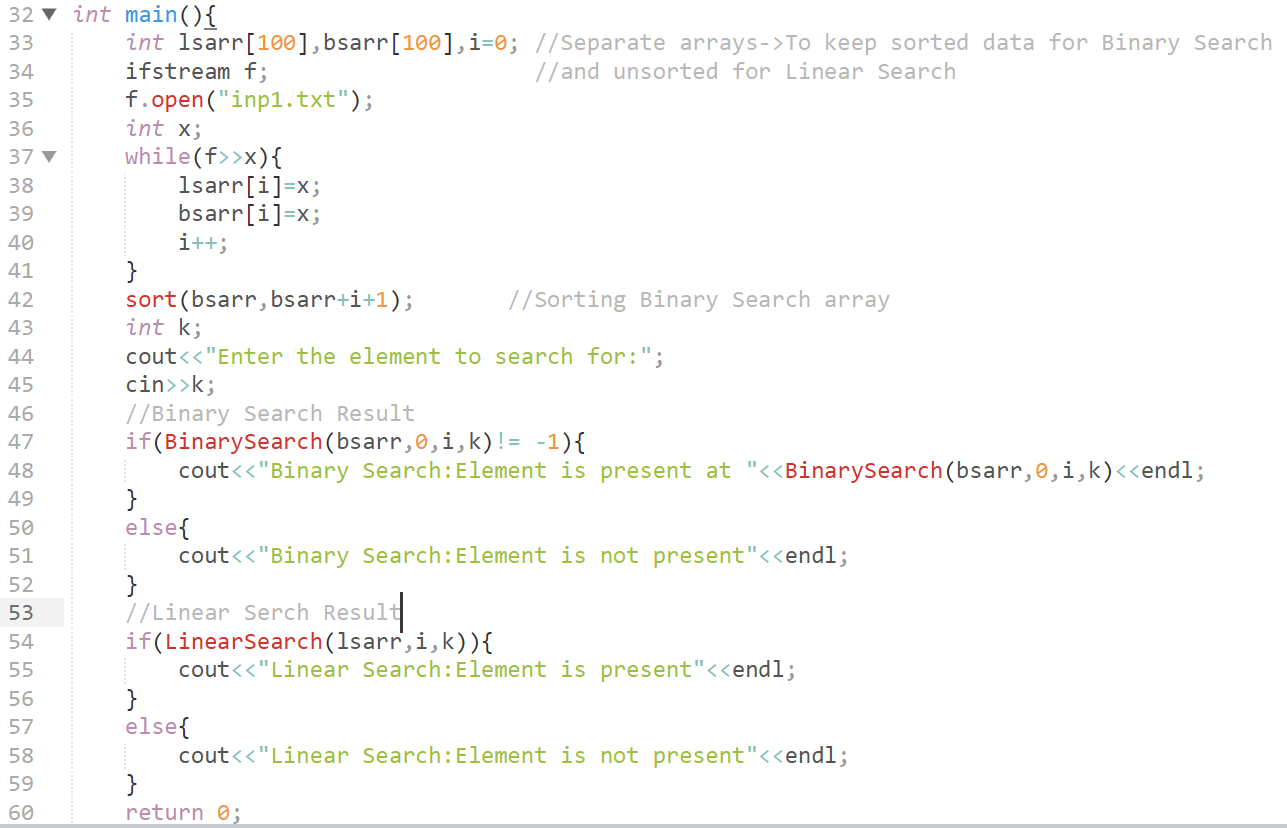
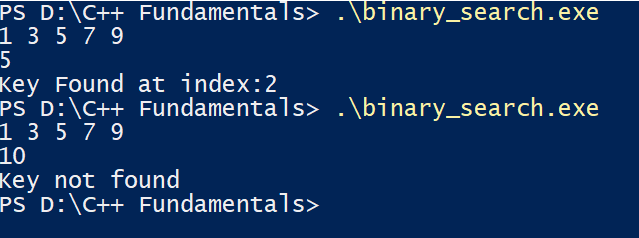
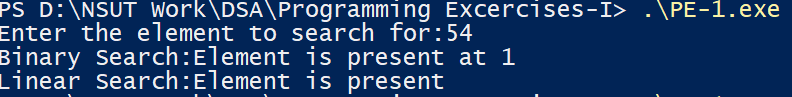
2021

**INDEX**

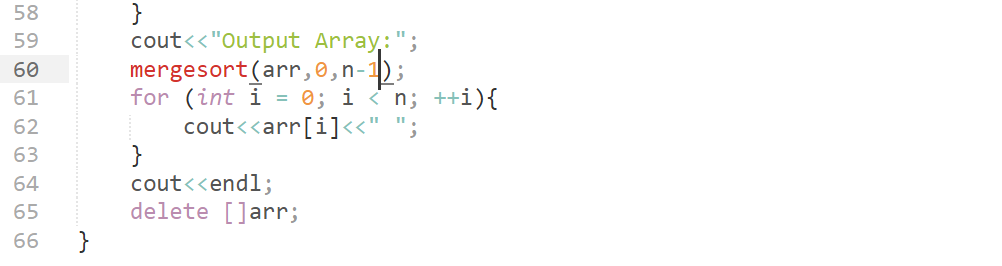
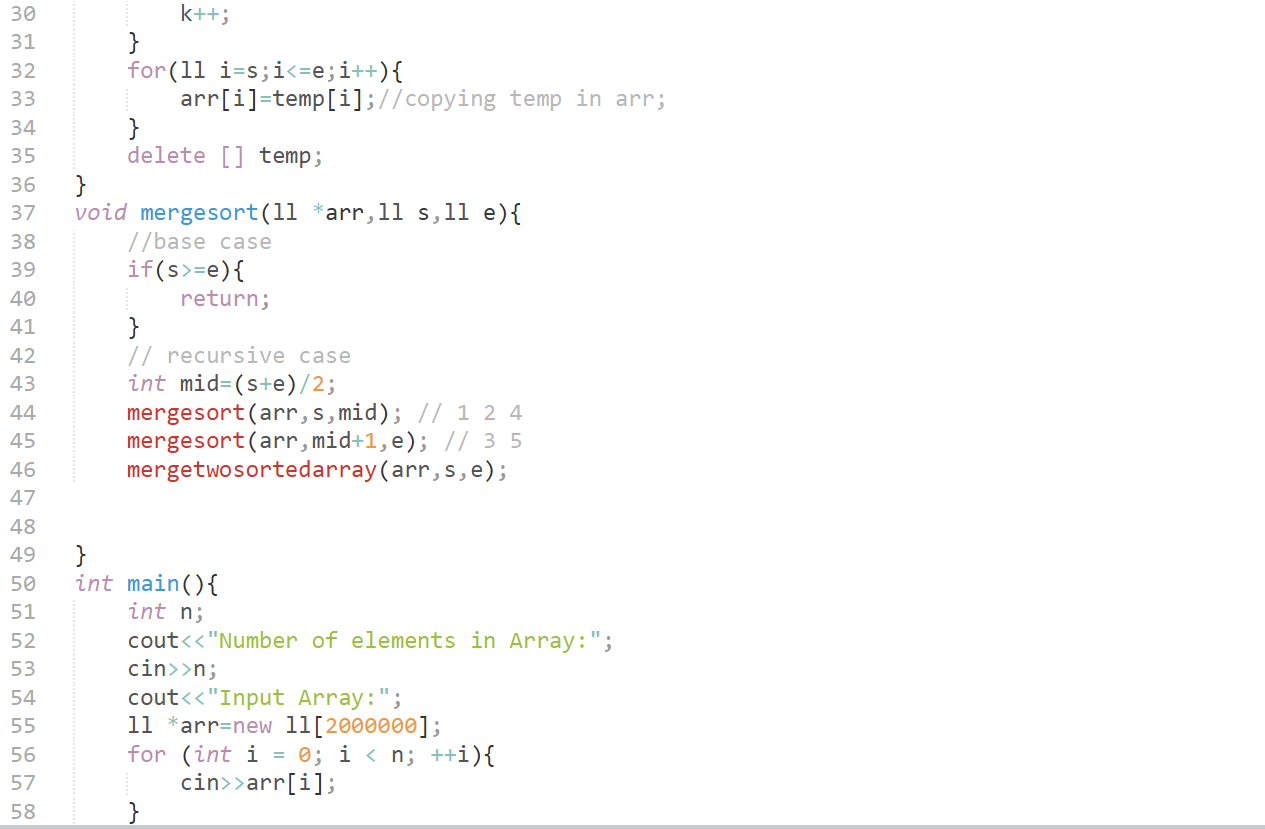
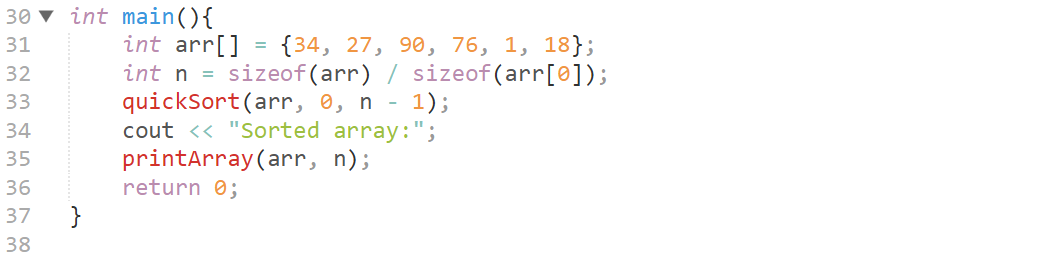
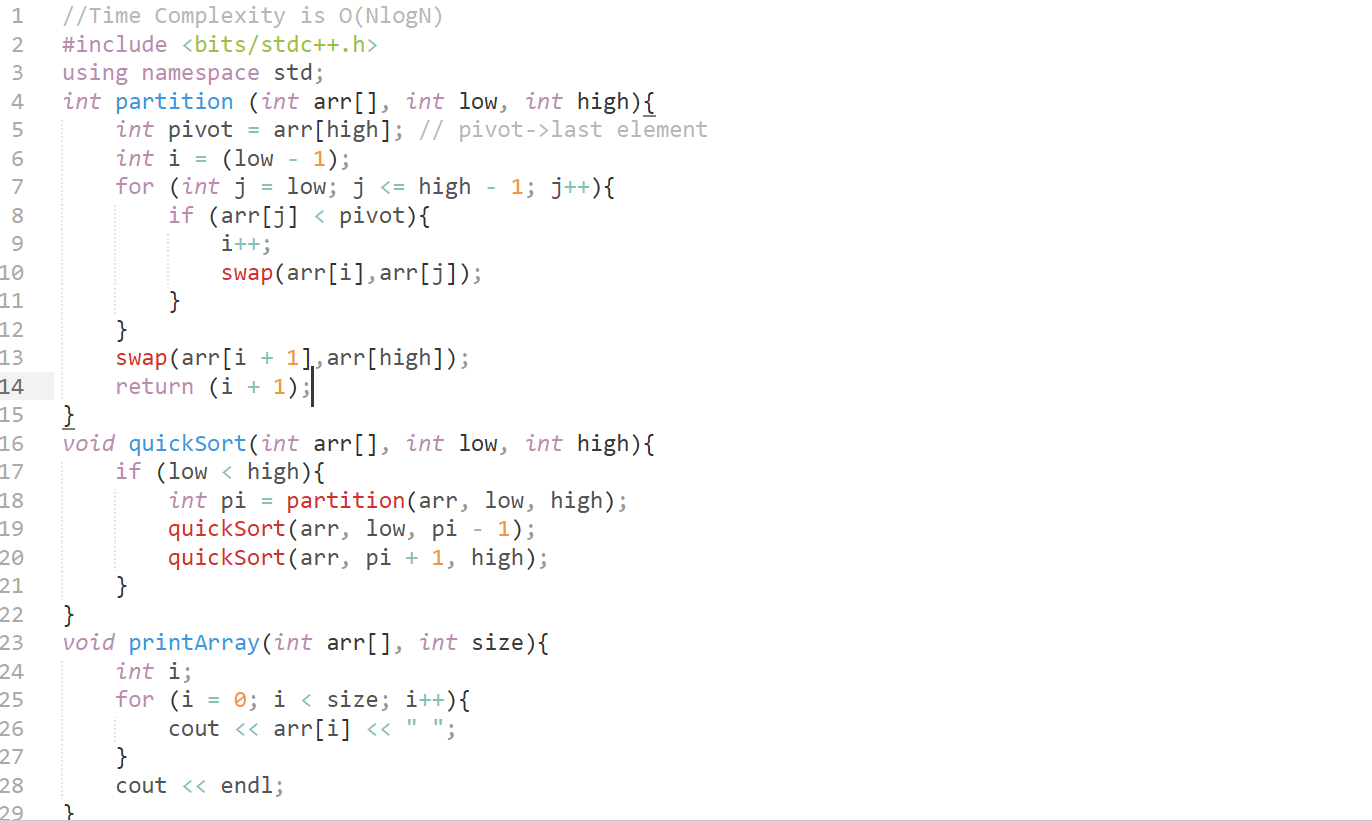
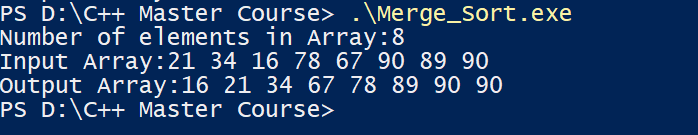
|  |  |  |
| --- | --- | --- |
| **SNO** | **LAB ASSIGNMENT** | **DATE** |
| 1 | Write a program for Iterative and Recursive Binary Search | 21.10.21 |
| 2 | Write a program for Merge Sort, Quick Sort | 22.10.21 |
| 3 | Write a program for Minimum Spanning Trees using Kruskal’s algorithm and Prim’s algorithm | 29.11.21 |
| 4 | Write a program for Floyd-Warshall algorithm. | 30.11.21 |
| 5 | Write a program for Single Source Shortest Path, write a program for Traveling salesman problem | 1.12.21 |
| 6 | Write a program for Optimal Merge Patterns, Write a program for Huffman Coding | 2.12.21 |
| 7 | Write a program for Hamiltonian Problem, Write a program for Strassen’s Matrix Multiplication | 3.12.21 |

**LAB-1**

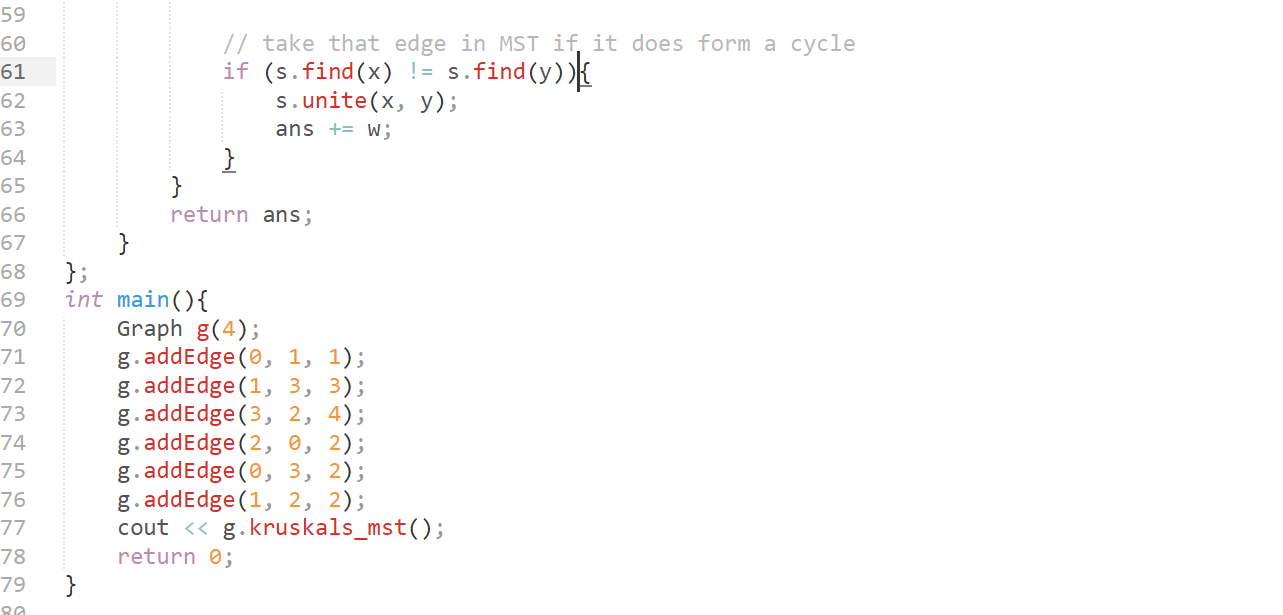
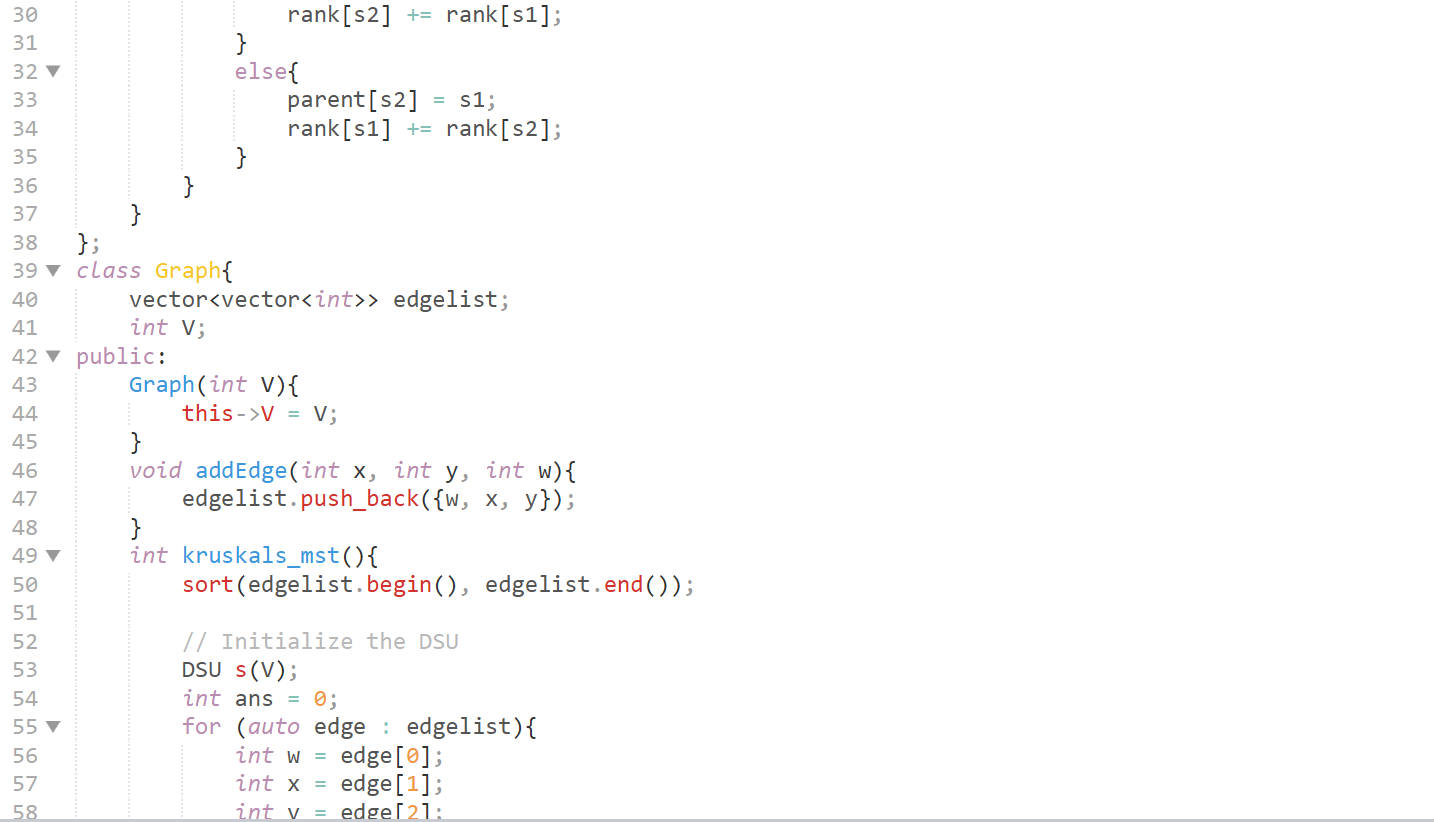
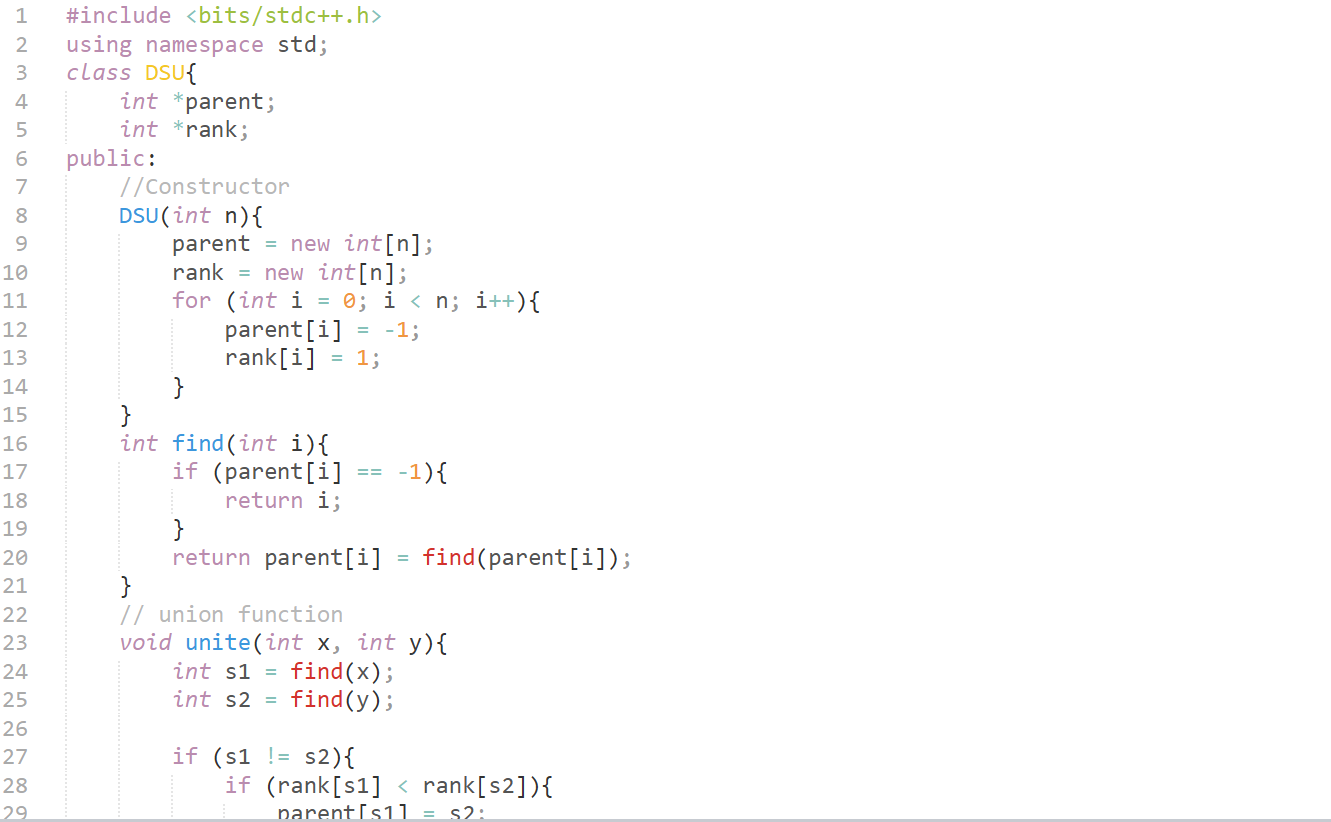
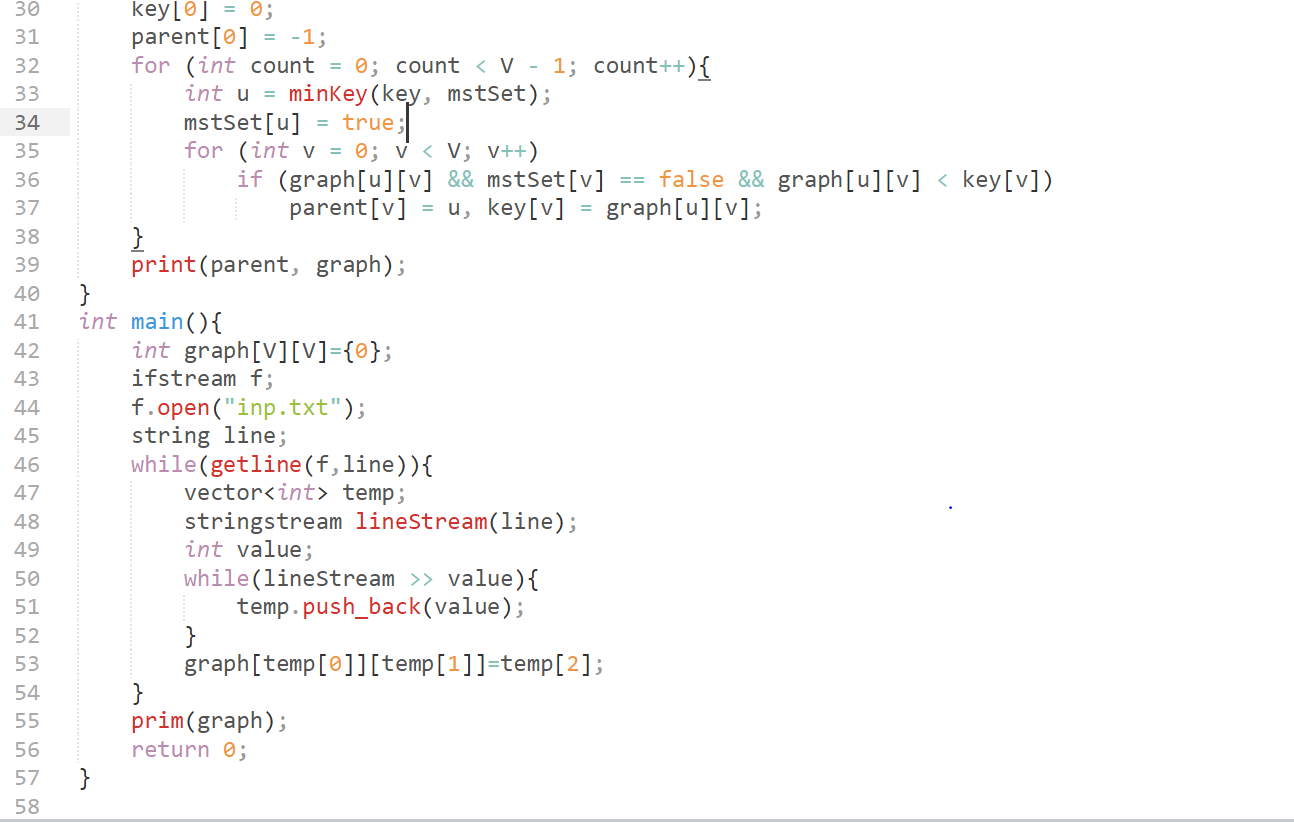
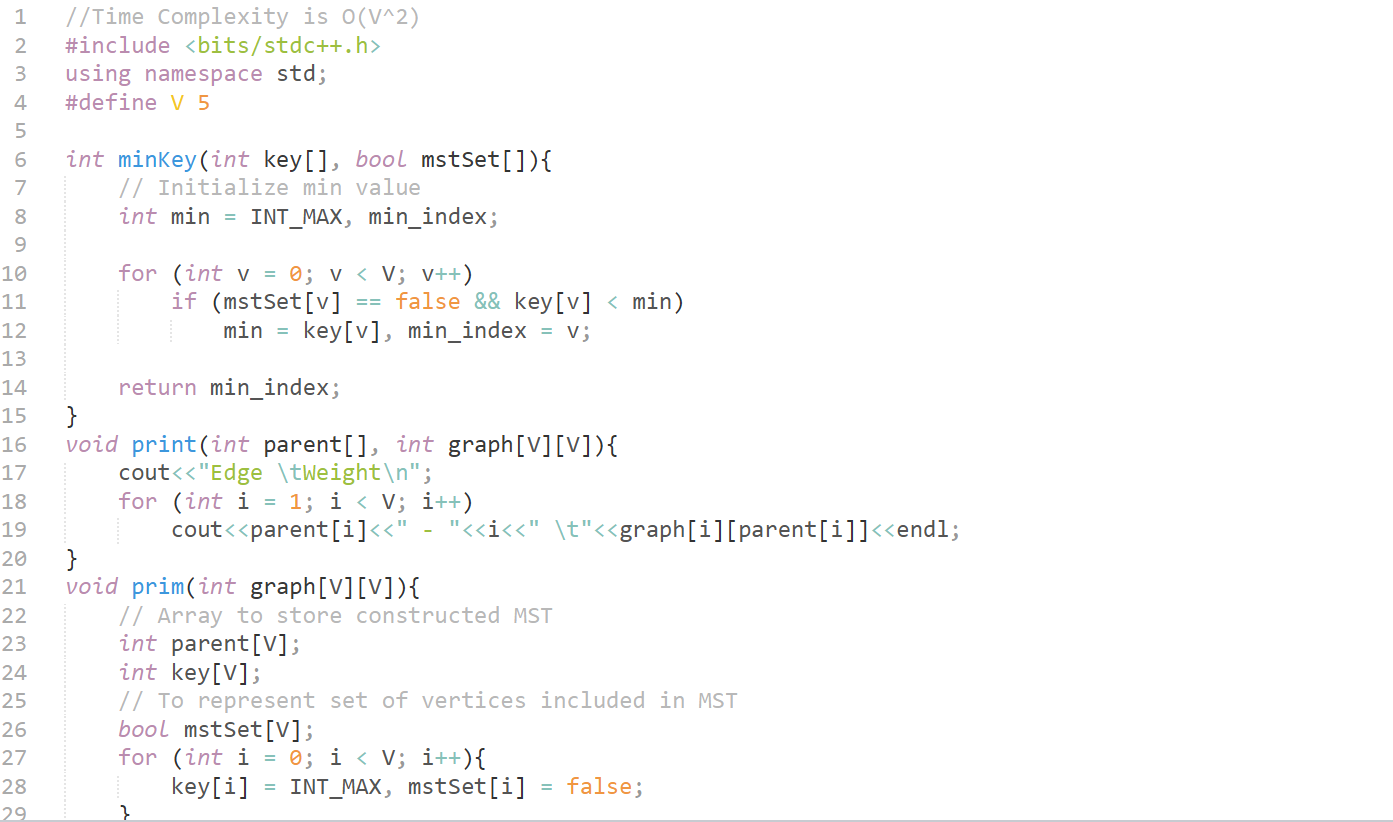
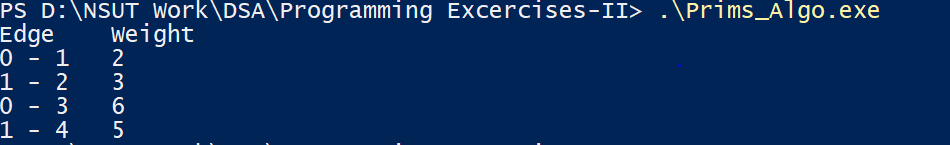
**CODE-ITERATIVE:**  **CODE-RECURSIVE:** 

 **OUTPUT-ITERATIVE:**  **OUTPUT-RECURSIVE:** 

**LAB-2**

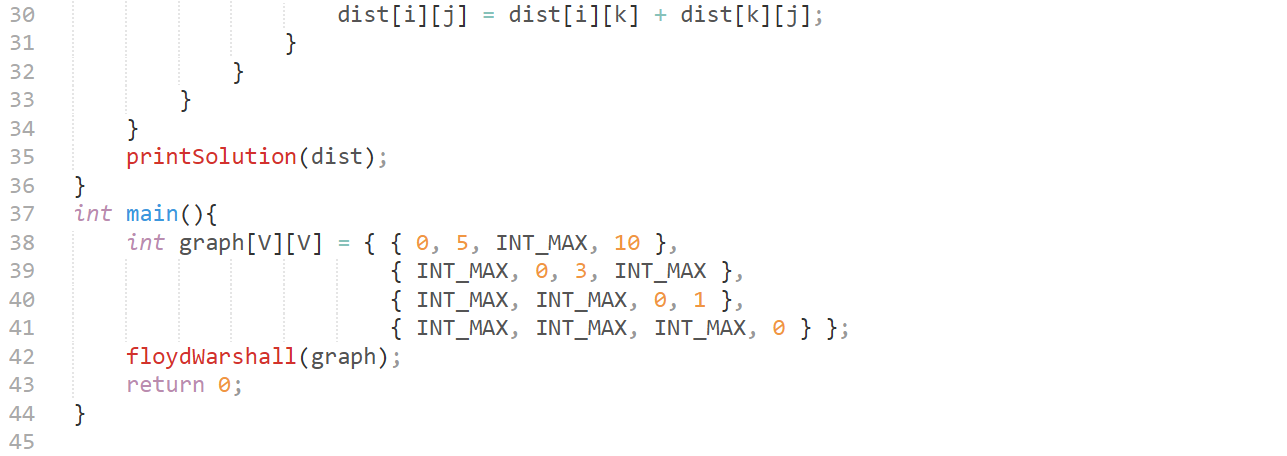
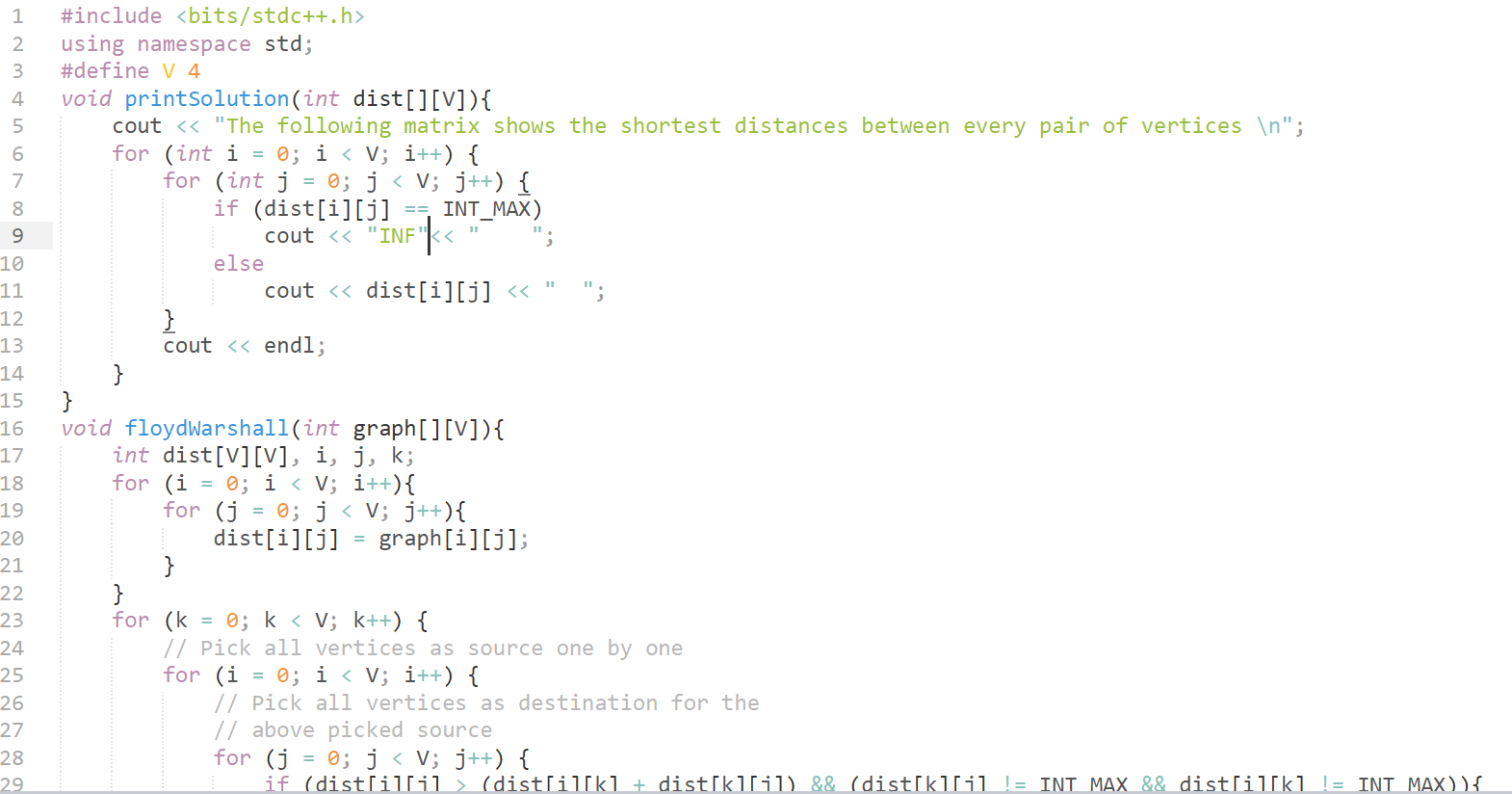
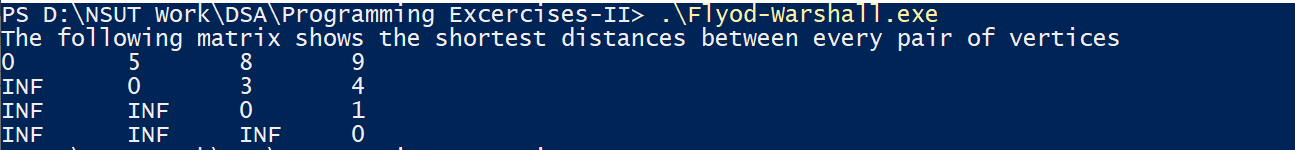
**CODE-MERGE SORT:**  **CODE-QUICK SORT:**  **OUTPUT-MERGE SORT:**  **OUTPUT-QUICK SORT:** 

**LAB-3**

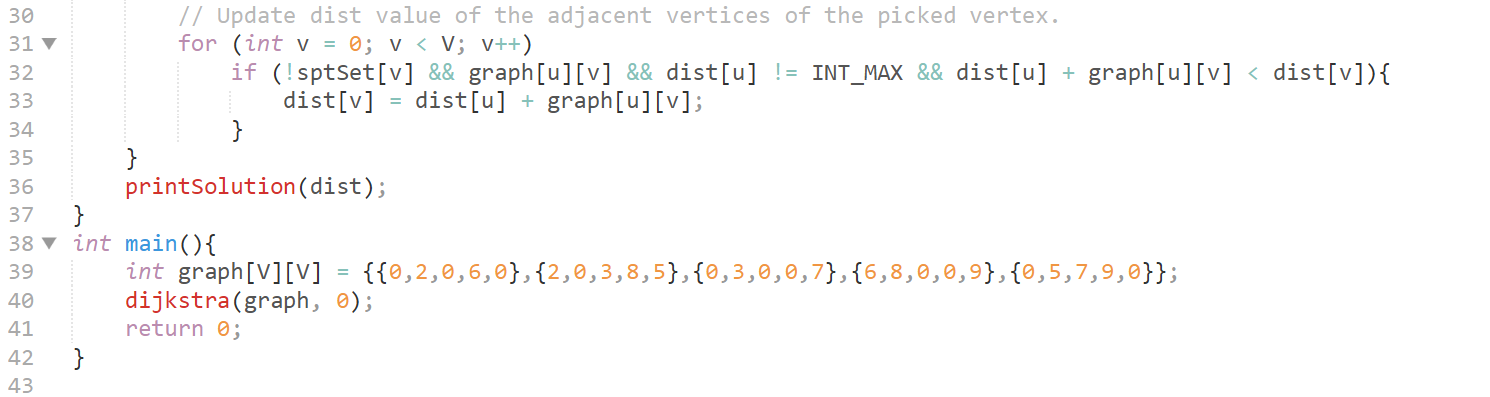
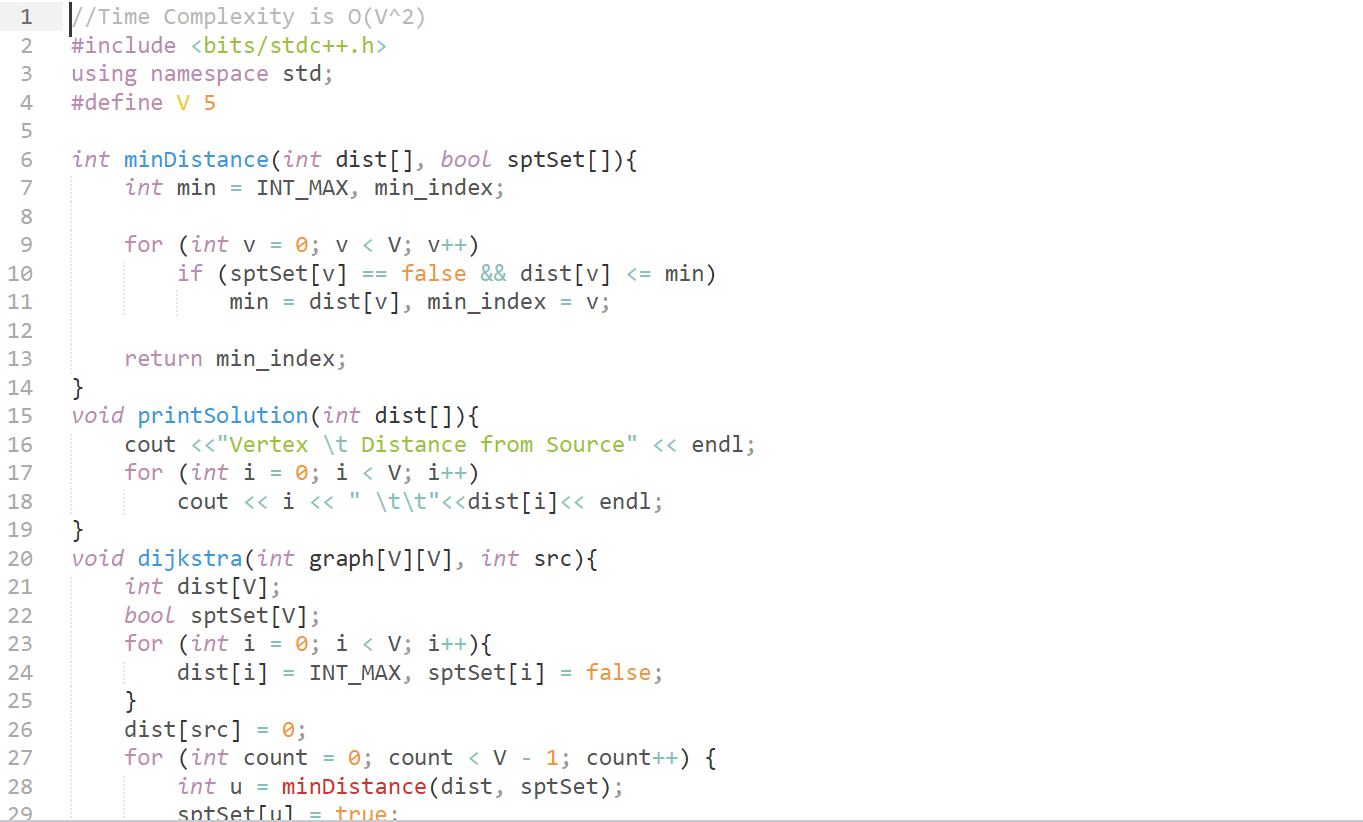
**CODE-KRUSKALS:**  **CODE-PRIMS:**  **OUTPUT-KRUSKALS:**  **OUTPUT-PRIMS:** 

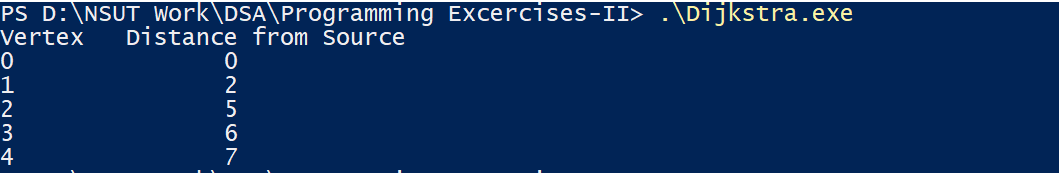
**LAB-4**

**CODE:**

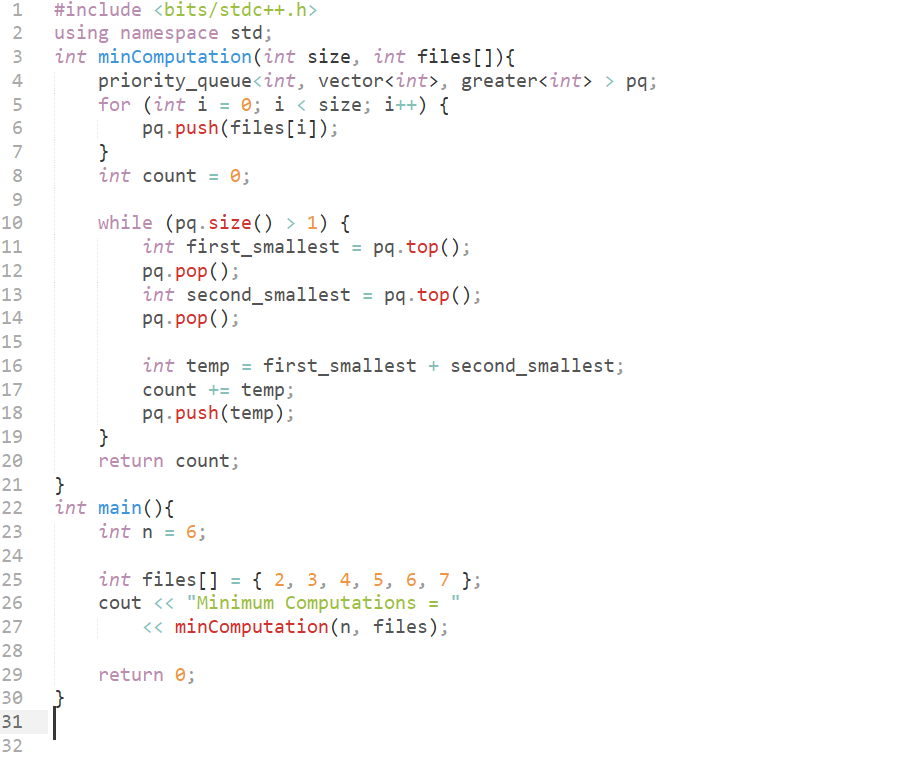
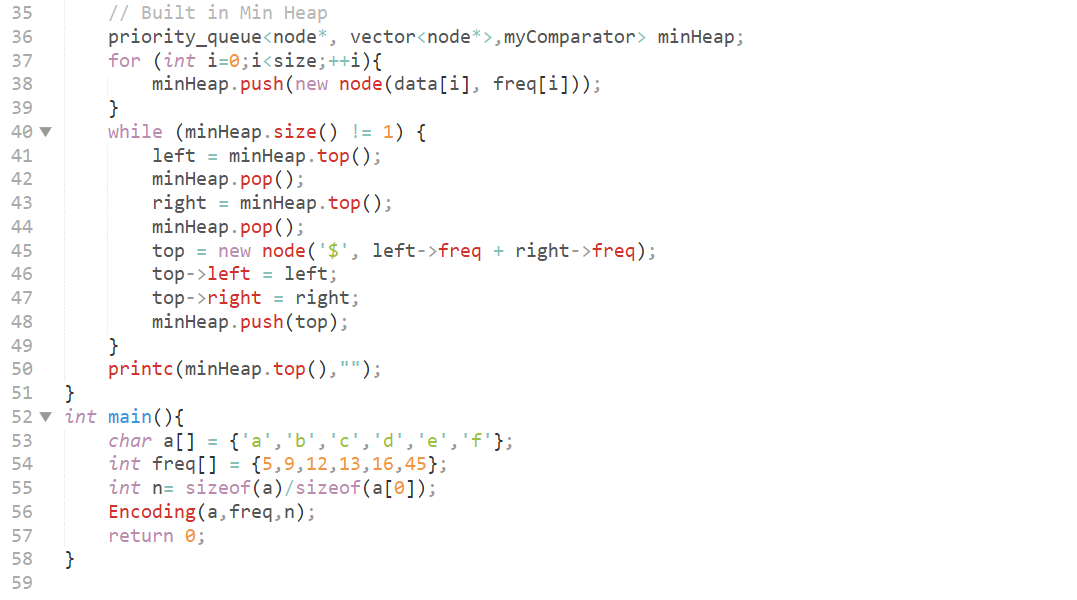
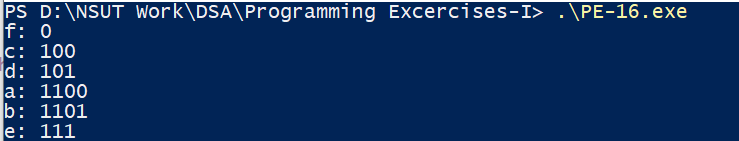
 **OUTPUT:** 

**LAB-5**

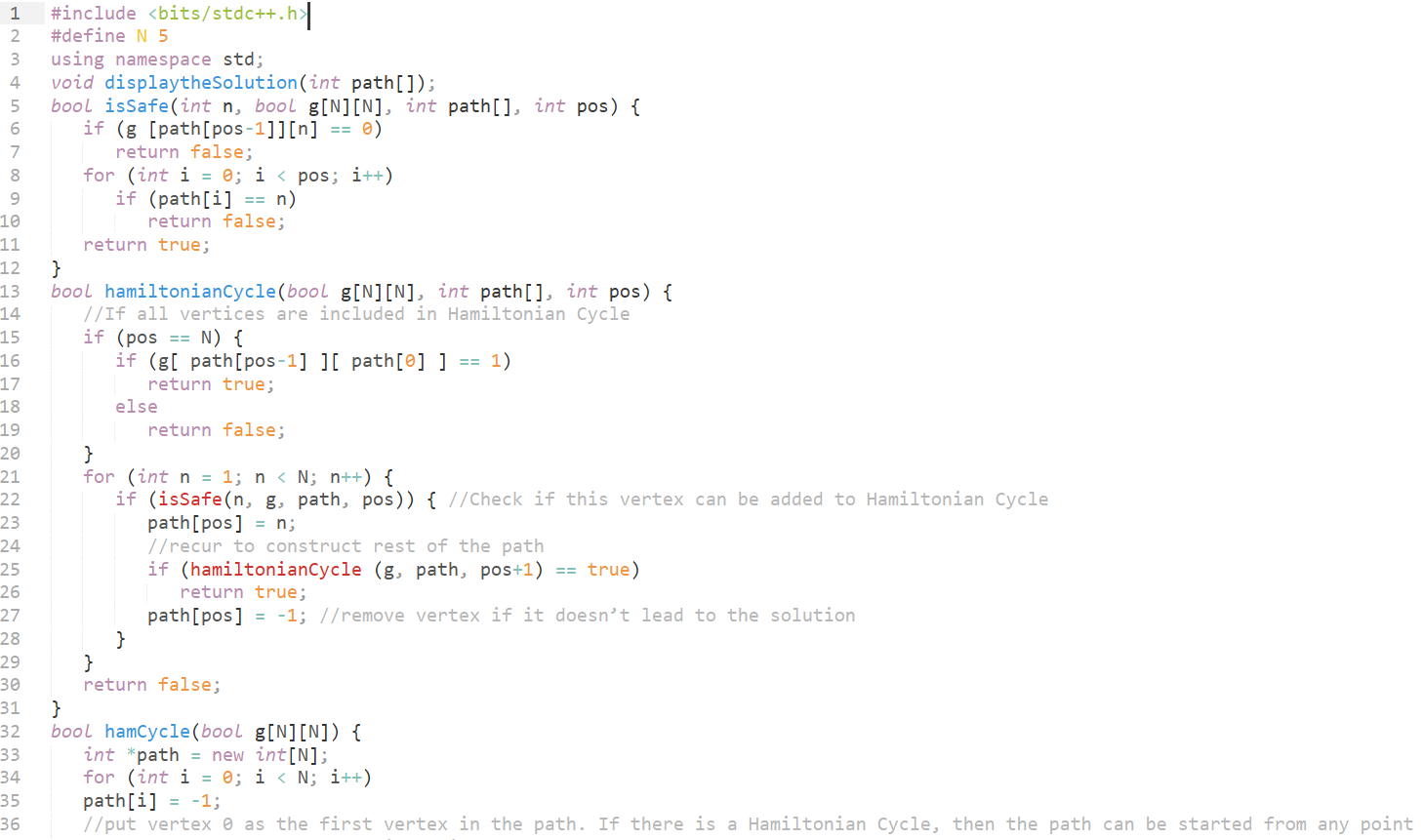
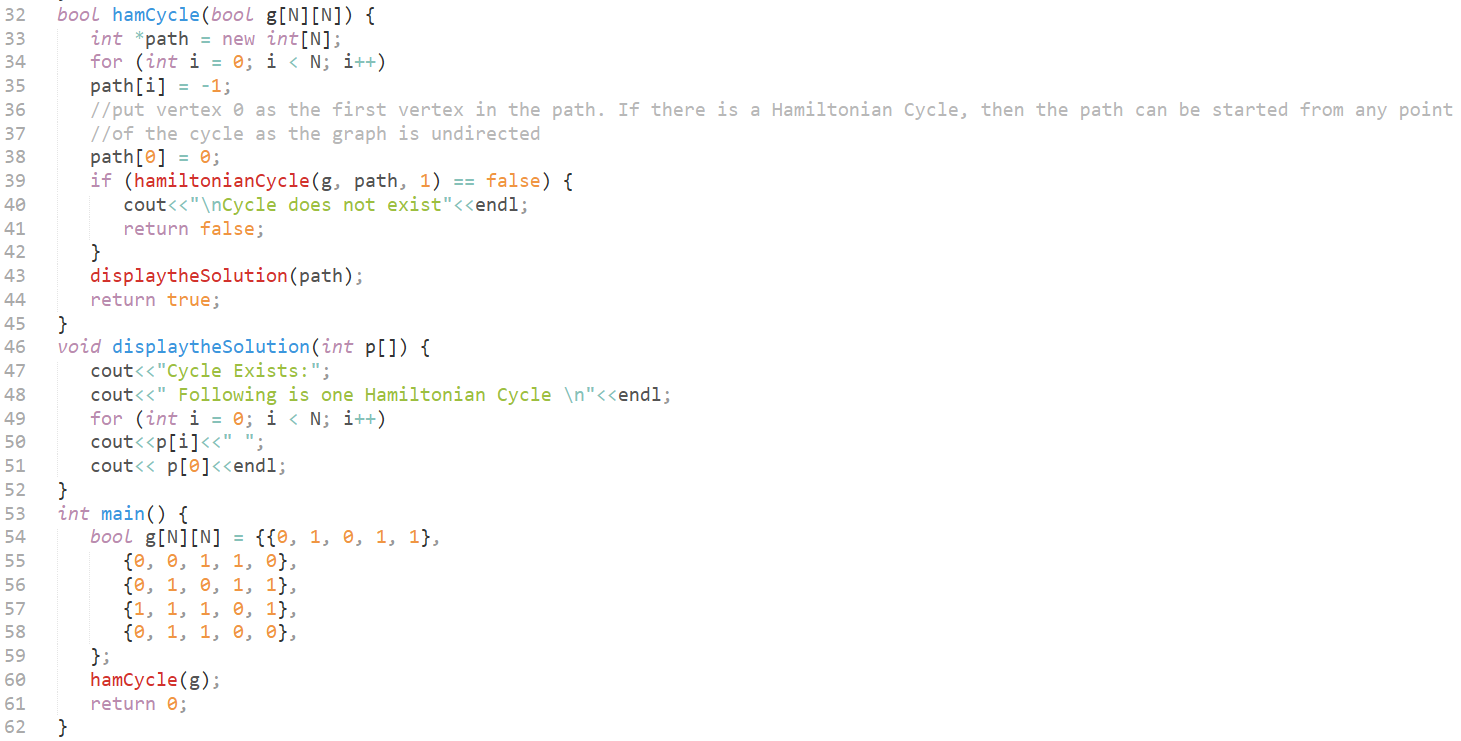
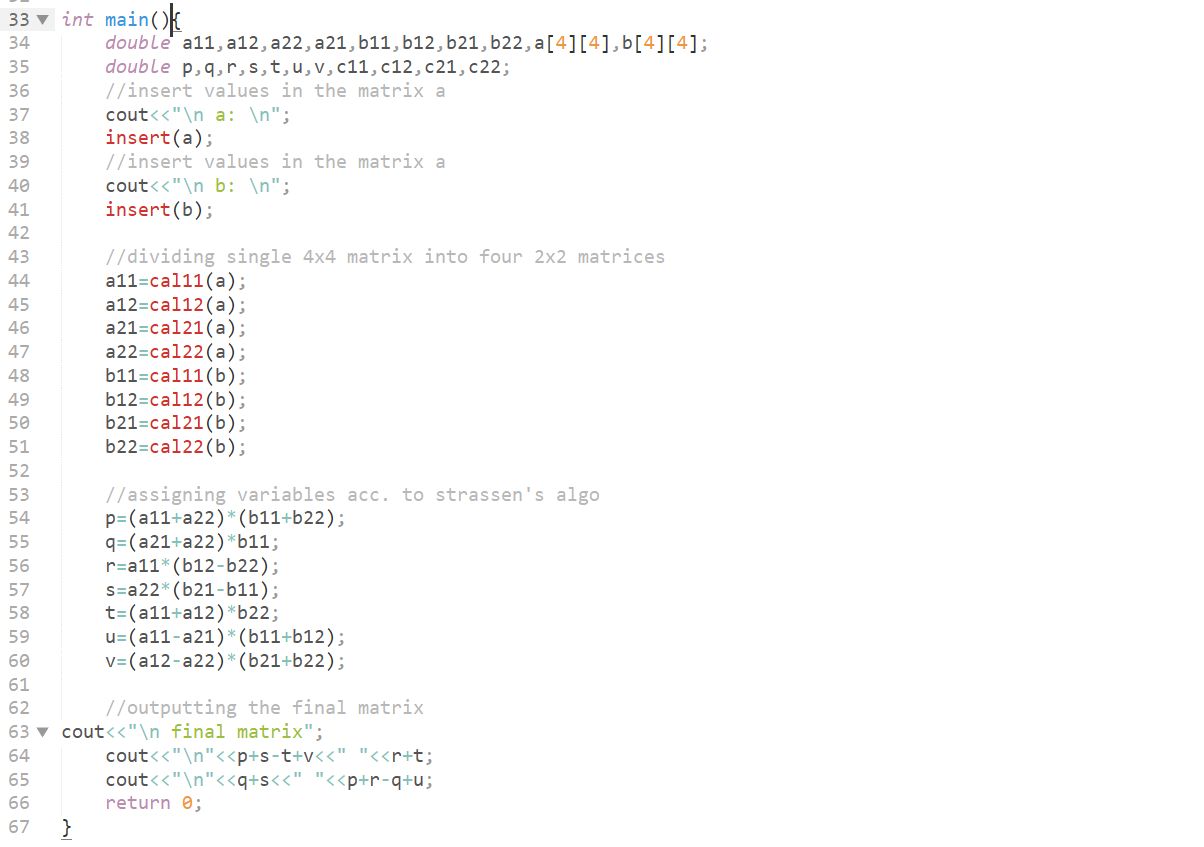
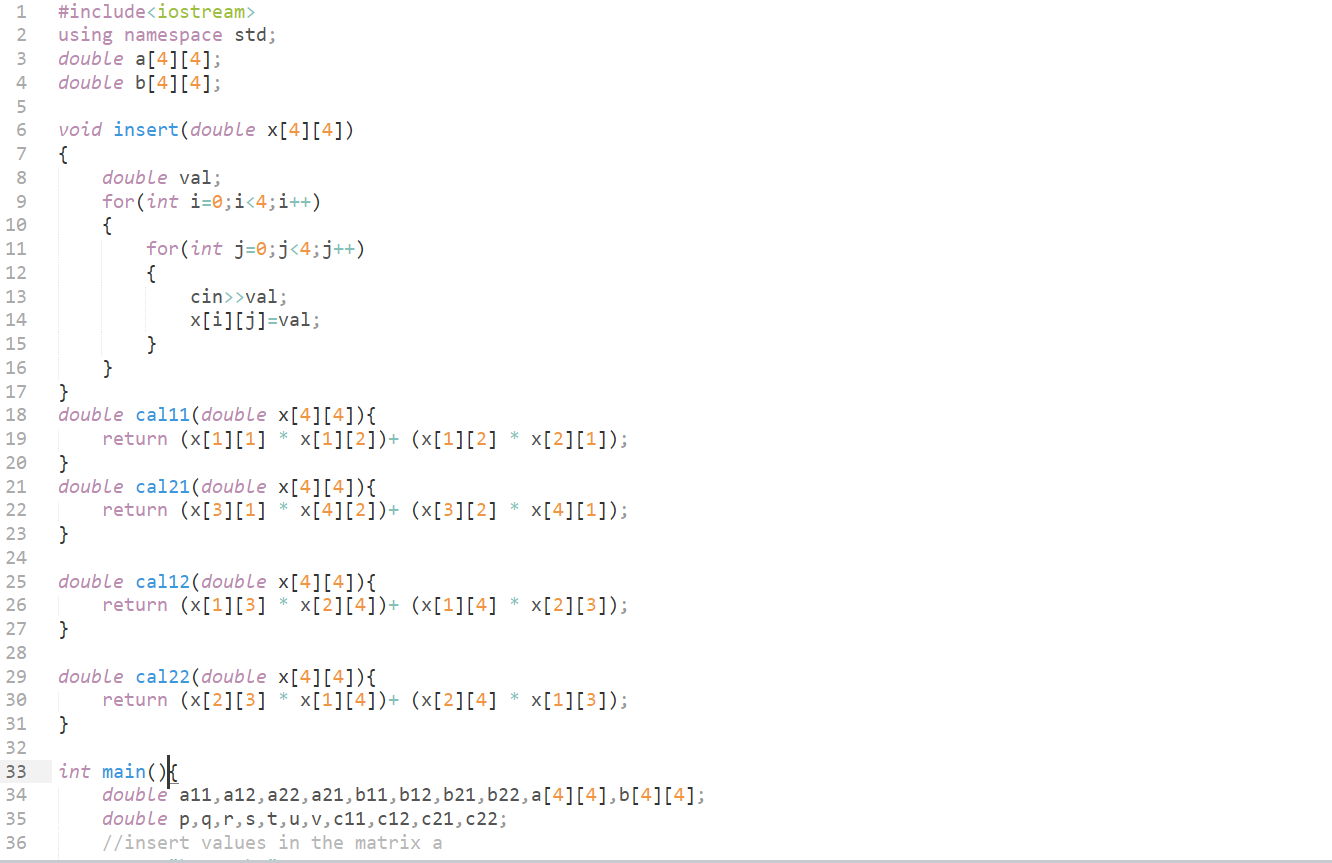
**CODE-DIJKSTRAS:** 

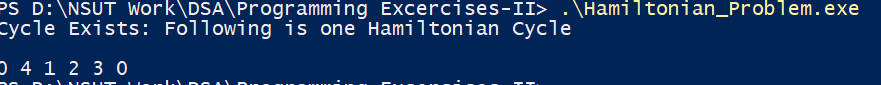
**CODE-TRAVELLING SALESMAN:**  **OUTPUT-DIJKSTRAS:**  **OUTPUT-TRAVELLING SALESMAN:** 

**LAB-6**

**CODE-OPTIMAL MERGE:**  **CODE-HUFFMAN ENCODING:**  **OUTPUT-OPTIMAL MERGE:**  **OUTPUT-HUFFMAN ENCODING:** 

**LAB-7**

**CODE-HAMILTONIAN PROBLEM:**   
 **CODE-STRASSENS MULTIPLICATION:** 

**OUTPUT-HAMILTONIAN PROBLEM:**  **OUTPUT-STRASSENS MULTIPLICATION:** 