1.quick sort

OUTPUT:

```
Unsorted Array
8 7 2 1 0 9 6
Sorted array in ascending order:
0 1 2 6 7 8 9
```

Merge sort

OUTPUT:

```
Array Before MergeSort
6 5 12 10 9 1
Sorted array:
1 5 6 9 10 12
```

2.

OUTPUT:

```
Sorted array:1 2 2 3 3 4 8
```

3.

OUTPUT:

```
1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:1
Enter the element to insert:78

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:1
Enter the element to insert:90

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:1
Enter the element to insert:45

3. Traverse     4. Exit
Enter your choice:1
Enter the element to insert:80

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:1
Enter the element to insert:08

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:3
```

```
Enter your choice:3
8 45 78 80 90

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:2
Enter the element to delete:90

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:3
8 45 78 80

1. Insertion    2. Deletion
3. Traverse     4. Exit
Enter your choice:4
PS C:\Users\10582\Desktop\KMCLU_ACADEMIC\5
```

4.

OUTPUT:

```
ENTER THE NUMBER OF ELEMENTS:4

ENTER THE ELEMENTS:
62
87
17
66

THE ROOT NODES ARE:-
17

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT
1

ENTER THE ELEMENT TO BE INSERTED:59

NOW THE HEAP IS:

THE ROOT NODES ARE:-
59-->17

INSERT MORE(N/Y)=
N
```

```
MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT
2

EXTRACTING THE MINIMUM KEY NODE
THE EXTRACTED NODE IS 17
NOW THE HEAP IS:

THE ROOT NODES ARE:-
59

EXTRACT MORE(N/Y)
N

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT
3

ENTER THE KEY OF THE NODE TO BE DECREASED:66

ENTER THE NEW KEY : 08

INVALID CHOICE OF KEY TO BE REDUCED
aster* ⊗ 0 ⚠ 0  0
```

```
NOW THE HEAP IS:

THE ROOT NODES ARE:-
59

DECREASE MORE(N/Y)
N

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT
4

ENTER THE KEY TO BE DELETED: 87

INVALID CHOICE OF KEY TO BE REDUCED
NODE DELETED SUCCESSFULLY
DELETE MORE(N/Y)
N

ENTER THE KEY TO BE DELETED: 87

INVALID CHOICE OF KEY TO BE REDUCED
NODE DELETED SUCCESSFULLY
DELETE MORE(N/Y)
N
ENTER THE KEY TO BE DELETED: 87

ENTER THE KEY TO BE DELETED: 87
```

```
NODE DELETED SUCCESSFULLY
DELETE MORE(N/Y)
N

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
NODE DELETED SUCCESSFULLY
DELETE MORE(N/Y)
N

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT

MENU:-

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
 4)DELETE A NODE
5)QUIT
5

1)INSERT AN ELEMENT
2)EXTRACT THE MINIMUM KEY NODE
3)DECREASE A NODEKEY
```

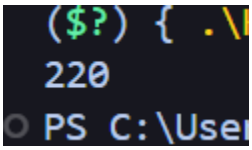
```
 4)DELETE A NODE
5)QUIT
5

5

PS C:\Users\10582\Desktop\KMCLU_ACADEMIC\5 semester\DAA>
```

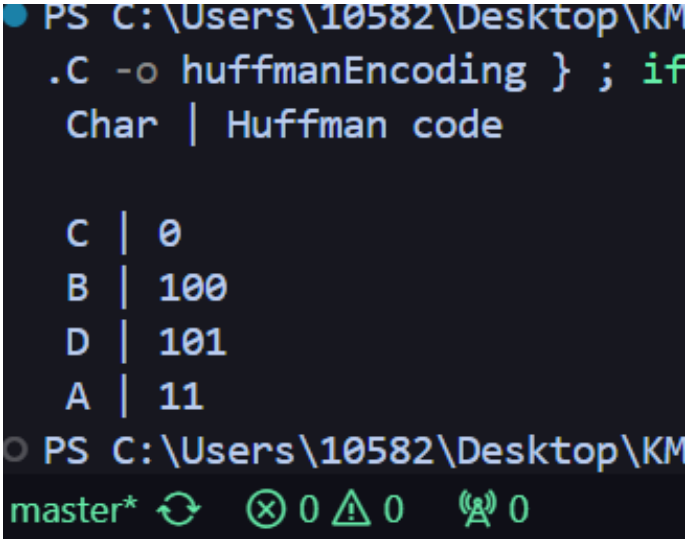
5.

OUTPUT:



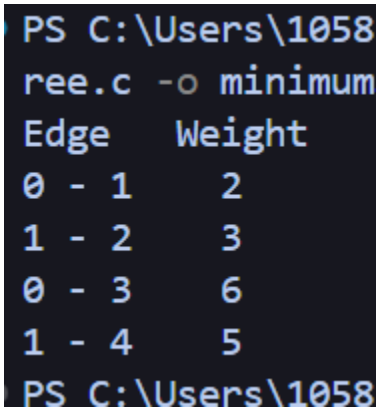
6.

OUTPUT:



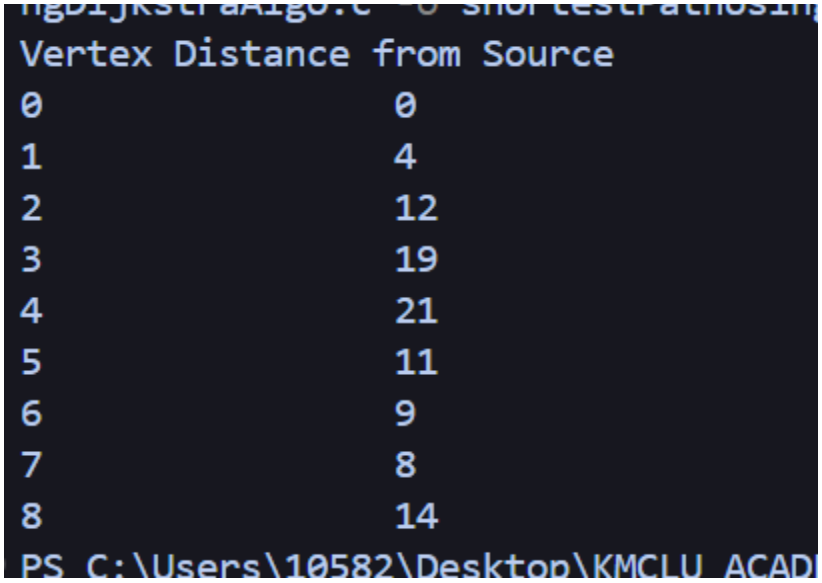
7.

OUTPUT:



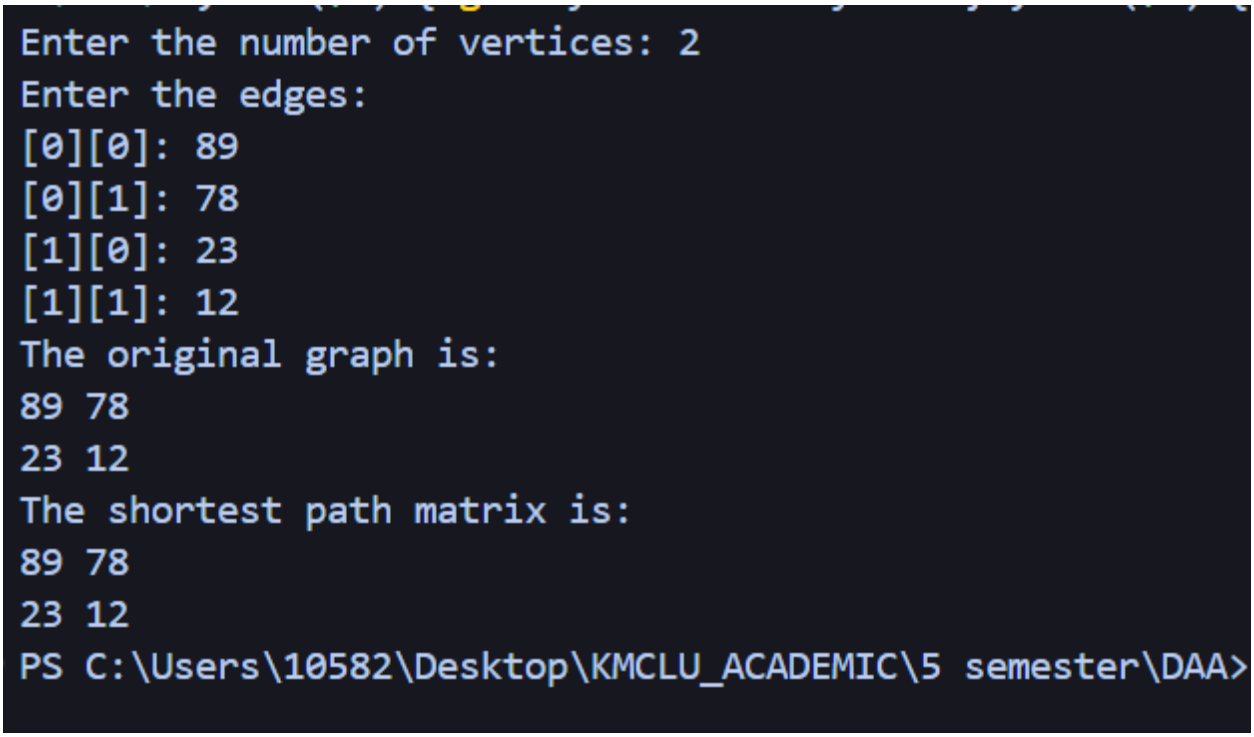
8.

OUTPUT:



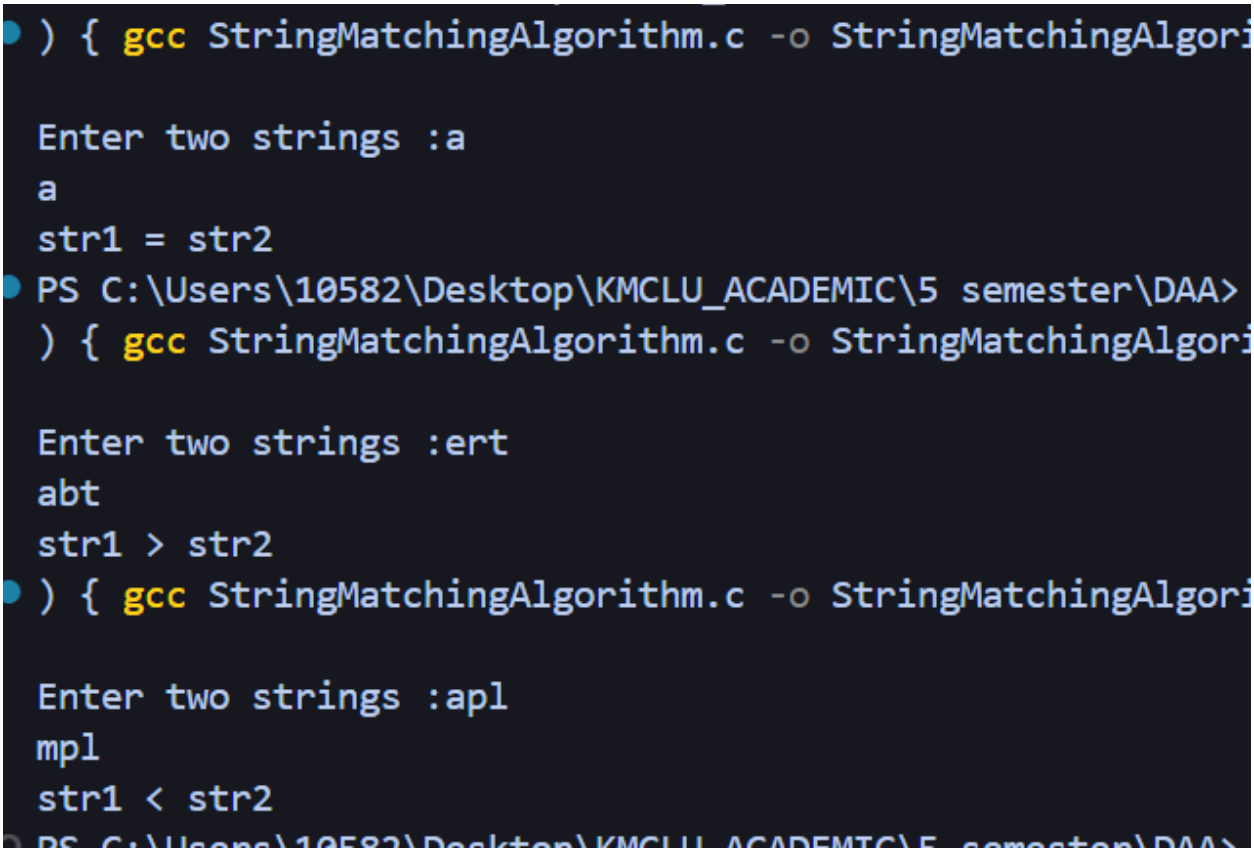
9.

OUTPUT:



10.

OUTPUT:


) { gcc StringMatchingAlgorithm.c -o StringMatchingAlgori
Enter two strings :a
a
str1 = str2
PS C:\Users\10582\Desktop\KMCLU_ACADEMIC\5 semester\DAA>
) { gcc StringMatchingAlgorithm.c -o StringMatchingAlgori
Enter two strings :ert
abt
str1 > str2
) { gcc StringMatchingAlgorithm.c -o StringMatchingAlgori
Enter two strings :apl
mpl
str1 < str2