**Experiment 1**

**AIM:** Write a program to merge 2 sorted arrays.

**Algo:**

Initialize x=0,y=0,a[100],b[100],c[100],i=0,j=0,k=0

read x

for id=0 to x-1 do:

read a[id];

for id=0 to y-1 do:

read b[id];

while i<x and j<y do:

if a[i]<b[j] then:

c[k]=a[i]

i=i+1

endif

if b[j]<a[i] then:

c[k]=b[j]

j=j+1

endif

k=k+1

endwhile

while i<x do:

c[k]=a[i]

i=i+1

k=k+1

endwhile

while j<y do:

c[k]=b[j]

j=j+1

k=k+1

endwhile

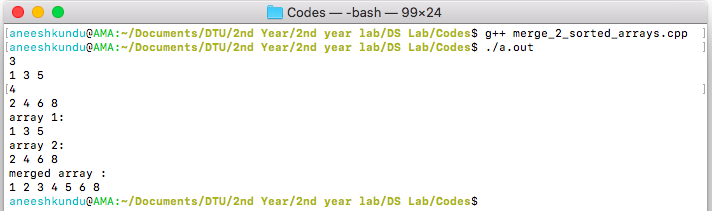
for id=0 to x+y-1 do:

write c[id]

**Code:**

1. #include < iostream >
2. using namespace std;
3. int a[10], b[10], c[20];
4. int x, y;
5. void merge(int \* a, int \* b, int \* c) {
6. int i = 0, j = 0, k = 0;
7. while (i < x && j < y) {
8. if (a[i] < b[j]) c[k++] = a[i++];
9. else c[k++] = b[j++];
10. }
11. while (i < x) c[k++] = a[i++];
12. while (j < y) c[k++] = b[j++];
13. }
14. int main() {
15. cin >> x;
16. for (int i = 0; i < x; i++)
17. cin >> a[i];
18. cin >> y;
19. for (int i = 0; i < y; i++)
20. cin >> b[i];
21. cout << "array 1:\n";
22. for (int i = 0; i < x; i++)
23. cout << a[i] << " ";
24. cout << endl;
25. cout << "array 2:\n";
26. for (int i = 0; i < y; i++)
27. cout << b[i] << " ";
28. cout << endl;
29. merge(a, b, c);
30. cout << "merged array :\n";
31. for (int i = 0; i < x + y; i++)
32. cout << c[i] << " ";
33. cout << endl;
34. return 0;
35. }

**Output:**



**Discussion:**

In this program, we use 2 pointers i and j to merge 2 sorted arrays. At each step we compare a[i] and b[j] and place the smaller one in the array c, and increment that pointer. When either of the 2 pointers reaches the end, we just copy the remaining elements of the other array into c.

**Conclusion:**

The program above merges 2 sorted arrays in where n and m are sizes of the sorted arrays.