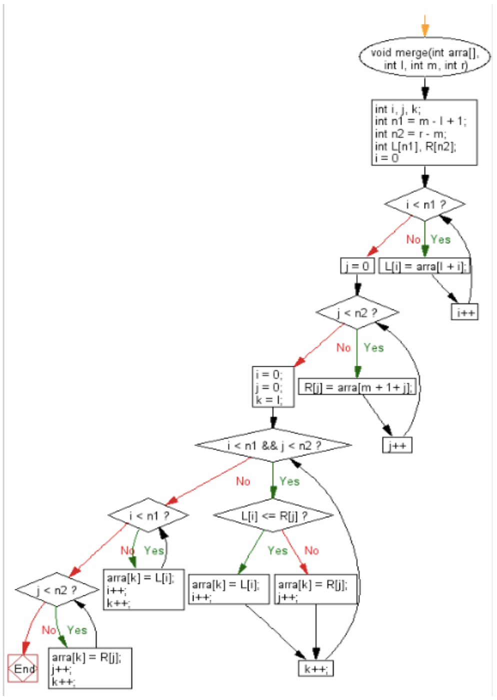
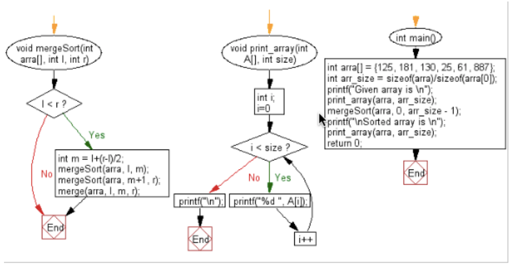
NAME – RAJDEEP JAISWAL	**DATE – 05 Oct 2021**
BRANCH – BTECH CSE	**SEC = 13 A**
UID -20BCS2761	
SUB- DS LAB	

Q -Write a C/C++ program to sort a list of elements using the merge sort algorithm.
Note: Merge sort is an O(n log n) comparison-based sorting algorithm. Most implementations produce a stable sort, which means that the implementation preserves the input order of equal elements in the sorted output.

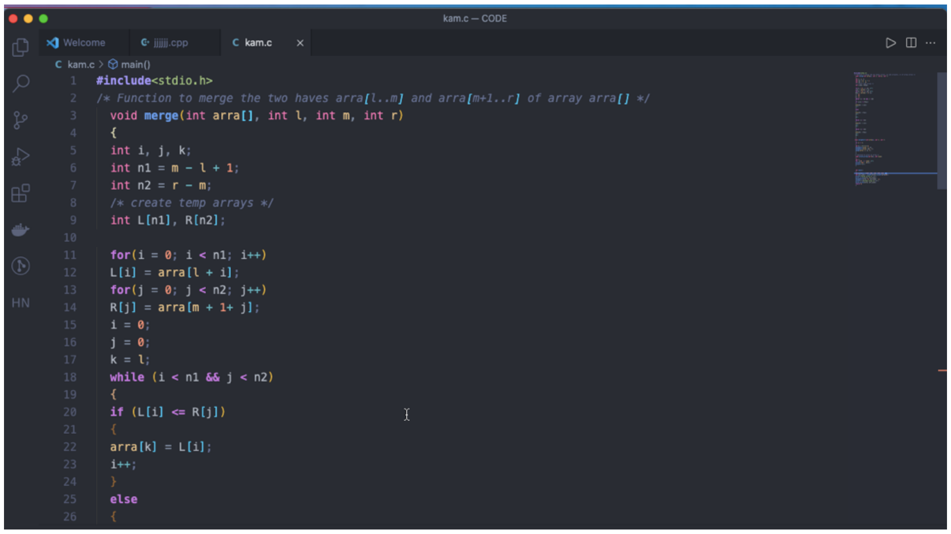
Solution –
FlowChart/Algorithms –



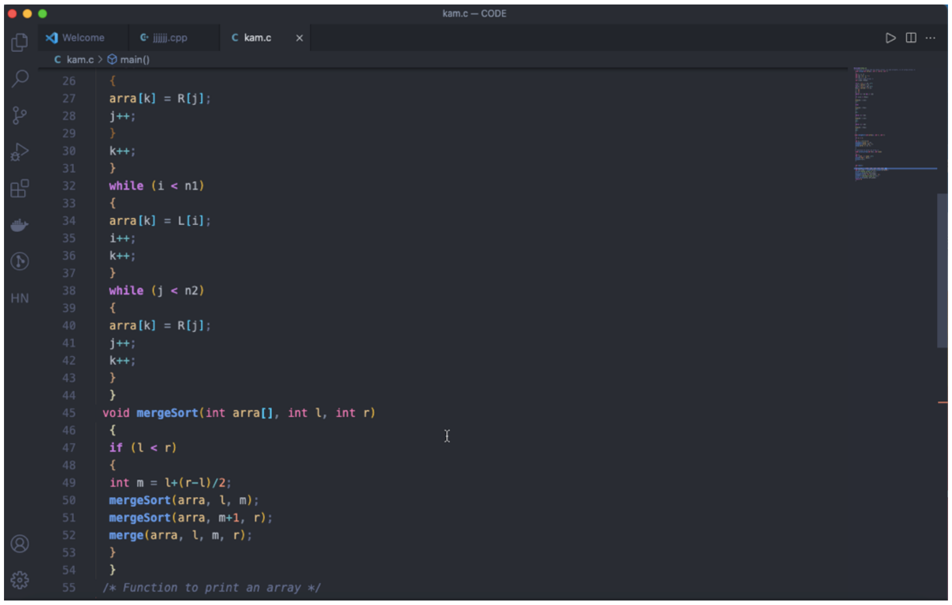

void mergeSort(int arra[], int l, int r)
l < r ?
Yes
No
int m = l+(r-l)/2; mergeSort(arra, l, m); mergeSort(arra, m+1, r); merge(arra, l, m, r);
End
void print_array(int A[], int size)
int i; i=0
i < size ?
No
Yes
printf("\n");
printf("%d ", A[i]);
End
i++
int main()
End

```
int arra[] = {125, 181, 130, 25, 61, 887};
int arr_size = sizeof(arra)/sizeof(arra[0]);
printf("Given array is \n");
print_array(arra, arr_size);
mergeSort(arra, 0, arr_size - 1);
printf("\nSorted array is \n");
print_array(arra, arr_size);
return 0;
```

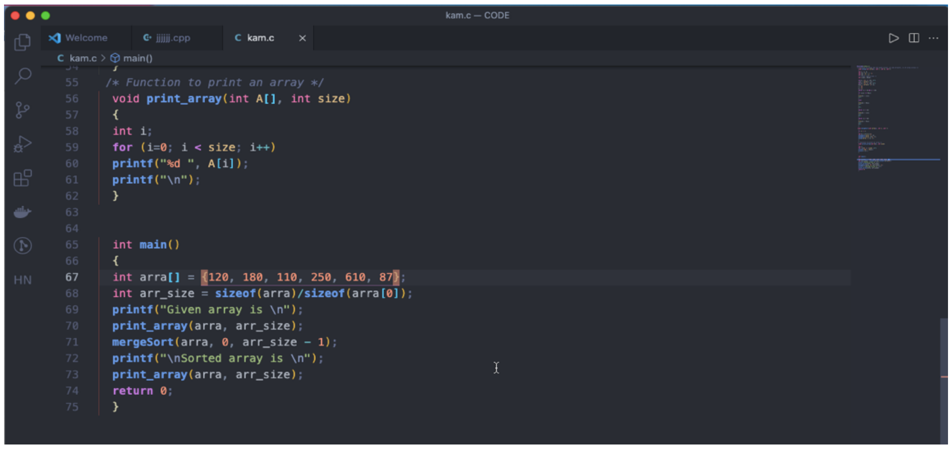
CODE IN COMPILER –



```c
#include<stdio.h>
/* Function to merge the two haves arra[l..m] and arra[m+1..r] of array arra[] */
  void merge(int arra[], int l, int m, int r)
  {
  int i, j, k;
  int n1 = m - l + 1;
  int n2 = r - m;
  /* create temp arrays */
  int L[n1], R[n2];

  for(i = 0; i < n1; i++)
  L[i] = arra[l + i];
  for(j = 0; j < n2; j++)
  R[j] = arra[m + 1+ j];
  i = 0;
  j = 0;
  k = l;
  while (i < n1 && j < n2)
  {
  if (L[i] <= R[j])
  {
  arra[k] = L[i];
  i++;
  }
  else
  {
```



```c
  {
  arra[k] = R[j];
  j++;
  }
  k++;
  }
  while (i < n1)
  {
  arra[k] = L[i];
  i++;
  k++;
  }
  while (j < n2)
  {
  arra[k] = R[j];
  j++;
  k++;
  }
  }
 void mergeSort(int arra[], int l, int r)
  {
  if (l < r)
  {
  int m = l+(r-l)/2;
  mergeSort(arra, l, m);
  mergeSort(arra, m+1, r);
  merge(arra, l, m, r);
  }
  }
 /* Function to print an array */
```



```c
 /* Function to print an array */
  void print_array(int A[], int size)
  {
  int i;
  for (i=0; i < size; i++)
  printf("%d ", A[i]);
  printf("\n");
  }


  int main()
  {
  int arra[] = {120, 180, 110, 250, 610, 87};
  int arr_size = sizeof(arra)/sizeof(arra[0]);
  printf("Given array is \n");
  print_array(arra, arr_size);
  mergeSort(arra, 0, arr_size - 1);
  printf("\nSorted array is \n");
  print_array(arra, arr_size);
  return 0;
  }
```

CODE IN TEXT –

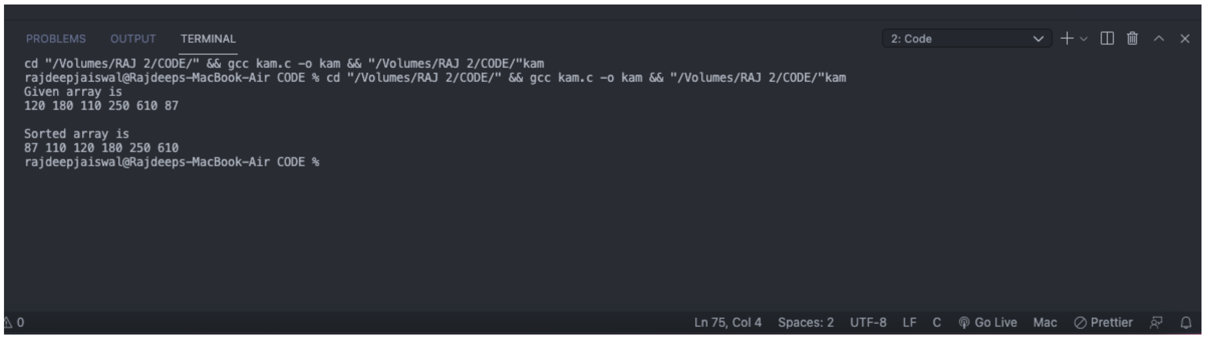
```
#include<stdio.h>

/* Function to merge the two haves arra[l..m] and arra[m+1..r] of array arra[] */
  void merge(int arra[], int l, int m, int r)
  {
  int i, j, k;
  int n1 = m - l + 1;
  int n2 = r - m;
  /* create temp arrays */
  int L[n1], R[n2];
for(i = 0; i < n1; i++)
  L[i] = arra[l + i];
  for(j = 0; j < n2; j++)
  R[j] = arra[m + 1+ j];
  i = 0;
  j = 0;
  k = l;
  while (i < n1 && j < n2)
  {
  if (L[i] <= R[j])
  {
  arra[k] = L[i];
  i++;
  }
  else
  {
  arra[k] = R[j];
  j++;
  }
  k++;
  }
  while (i < n1)
  {
  arra[k] = L[i];
  i++;
  k++;
  }
  while (j < n2)
  {
  arra[k] = R[j];
  j++;
  k++;
  }
  }
 void mergeSort(int arra[], int l, int r)
  {
  if (l < r)
  {
  int m = l+(r-l)/2;
  mergeSort(arra, l, m);
  mergeSort(arra, m+1, r);
  merge(arra, l, m, r);
  }
  }
 /* Function to print an array */
  void print_array(int A[], int size)
  {
  int i;
  for (i=0; i < size; i++)
  printf("%d ", A[i]);
  printf("\n");
  }
int main()
  {
  int arra[] = {120, 180, 110, 250, 610, 87};
  int arr_size = sizeof(arra)/sizeof(arra[0]);
  printf("Given array is \n");
  print_array(arra, arr_size);
  mergeSort(arra, 0, arr_size - 1);
  printf("\nSorted array is \n");
  print_array(arra, arr_size);
  return 0;
  }
```

CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

OUTPUT -

GIVEN ARRAYS IS - {120, 180, 110, 250, 610, 87};

SORTED ARRAY IS – 87 110 120 180 250 610



```
cd "/Volumes/RAJ 2/CODE/" && gcc kam.c -o kam && "/Volumes/RAJ 2/CODE/"kam
rajdeepjaiswal@Rajdeeps-MacBook-Air CODE % cd "/Volumes/RAJ 2/CODE/" && gcc kam.c -o kam && "/Volumes/RAJ 2/CODE/"kam
Given array is
120 180 110 250 610 87

Sorted array is
87 110 120 180 250 610
rajdeepjaiswal@Rajdeeps-MacBook-Air CODE %
```