Lab Programs

ch. karthik
AP19110010529
CSE-H

1) Wap for insert sort algorithm

```
#include<stdio.h>
void main()
{
  int n, array[1000], c, d, t;
  printf("Enter no of elements \n");
  scanf("%d", &n);
  Printf("Enter %d integers \n", n);
  for (c=0, c<n; c++){
      Scanf("%d ", &array[c]);
  }
  for(c = 1; c<=n; c++){
     d = c;
     while (d>0 && array[d-1]>array[d]){
          t = array[d];
          array[d] = array[d-1];
          array[d-1] = t;
          d--;
        }
    }
    Printf("Sorted array in ascending order: \n");
    for(c=0; c<=n-1; c++){
       printf("%d\n", array[c]);
}
```

Output

```
Enter no of Elements
10
Enter 10 digits
10
9
8
7
6
5
4
3
2
1
Sorted array in ascending order
1
2
3
4
5
6
7
8
9
10
```

2) Write a program for the selection sort

```
#include <stdio.h>
void main()
{
    int array[1000], n, a, b, position, temp;
    printf("Enter no of elements\n");
    scanf("%d", &n);
    printf("Enter %d integers\n", n);
    for(a=0; a<n; a++){
        scanf("%d", &array[a]);
    }
    for(a=0; a<(n-1); a++){
        position = a;
        for(b=a+1; b<n; b++){
    if(array[position] > array[b])
        position = b;
    }
    if(position != a){
        temp = array[a];
        array[a] = array[position];
        array[position] = temp;
    }
    }
    Printf("Sorted array in ascending order:\n");
    for(a=0; a<n; a++){
        Printf("%d\n", array[a]);
    }
}
```

Output

```
Enter no of Elements
10
Enter 10 digits
10
9
8
7
6
5
4
3
2
1
Sorted array in ascending order
1
2
3
4
5
6
7
8
9
10
```

3) wap for Bubble sort Algorithm

```
#include<stdio.h>
void main()
{
  int array[1000] ; x, a, b, position, temp;
  printf ("Enter no of elements \x");
  scanf ("%d", &x);
  Printf ("Enter %d integers \x", x);
  for (a=0; a<x ; a++){
      scanf ("%d", &array [a]);
  }
  for (a=0 ; a<(x-1); a++) {
      for (b=0 ; b<x-a-1 ; b++) {
          if (array[b] > array[b+1]){
          temp = array [b];
          array [b] = array [b++];
          array [b+a] = temp;
          }
      }
  }
  printf("Sorted list Asending order: \x")
  for(a=0; a<x; a++){
      Printf (" %d\x", array[a]);
  }
}
```

Output

```
Enter no of Elements
10
Enter 10 digits
10
9
8
7
6
5
4
3
2
1
Sorted array in ascending order
1
2
3
4
5
6
7
8
9
10
```

4) Wap for the Merge Sort algorithm

```
#include<stdlib.h>
#include<stdio.h>
void merge(int arr[], int a, int b, int c)
{
  int x, y, z;
  int d1 = b - a + 1;
  int d2 = c - b;
  int a[d1], c[d2];
  for (x=0; x<d1; x++)
    a[x] = arr[a+x];
  for (y=0; y<d2; y++)
    c[y] = arr[b+a+y];
  x=0;
  y=0;
  z=a;
  while (x<d1 && y<d2)
  {
    if (a[x] <= c[y])
    {
      arr[z] = a[x];
      x++;
    }
```

```
else
{
    array[z] - c[y];
    y++;
  }
  z++;
}
while(x<d1)
{
  arr[z] = a[x];
      z++;
      z++;
while (y<d2)
{
  arr[z] = a[y];
    y++;
    z++;
  }
}
void mergesort(int arr[], int a, int c)
    {if (a<c)
      {int b = a+(c-1)/2;
       mergesort(arr, a, c);
       mergesort(arr, b+1, c);
       merge(arr, a, b, c);
    }
}
```

```
Void printArray(int I[ ], int SIZE)
{
    int x;
    for (x=0; I<SIZE; x++)
    printf("%d", I[I]);
    Printf("\d");
}
int main()
{
    int arr[] = {9, 2, 8, 4, 5, 6};
    int arr_SIZE = SIZEof(arr)/SIZEof(arr[0]);
    printf("Given array is \d");
    PritArray(arr, arr-SIZE);
    mergeSort(arr, 0, arr-SIZE);
    Printf("\d sorted array is \d");
    Print Array(arr, arrSIZE);
    return 0;
}
```

Output

Sorted array is

2, 4, 5, 6, 8, 9