NAME:CH.JYOTHIRMAI

ROLL NO:CH.SC.U4CSE24107

DAA WEEK-2

1.Bubble Sort

Code:

```c
//bubble sort
#include <stdio.h>
int main() {
int a[] = {5, 2, 8, 1, 3};
int n = 5;
int i, j, temp;
for(i = 0; i < n - 1; i++) {
for(j = 0; j < n - i - 1; j++) {
if(a[j] > a[j + 1]) {
temp = a[j];
a[j] = a[j + 1];
 a[j + 1] = temp;
 }
}
}
printf("Bubble Sorted: ");
for(i = 0; i < n; i++)
printf("%d ", a[i]);
return 0;
}
```

Output:

```
0 0 256 amma@amma05:~/Documents$ gcc bubble.c -o bubble
amma@amma05:~/Documents$ ./bubble
Bubble Sorted: 1 2 3 5 8 amma@amma05:~/Documents$
```

2.Insertion Sort

Code:

```c
//insertion sort
#include <stdio.h>
int main() {
    int a[] = {5, 2, 8, 1, 3};
    int n = 5;
    int i, key, j;
    for(i = 1; i < n; i++) {
        key = a[i];
        j = i - 1;
        while(j >= 0 && a[j] > key) {
            a[j + 1] = a[j];
            j--;
        }
        a[j + 1] = key;
    }
    printf("Insertion Sorted: \n");
    for(i = 0; i < n; i++)
        printf("%d ", a[i]);
    return 0;
}
```

Output:

```
n
amma@amma05:~/Documents$ ./insertion
Insertion Sorted:
1 2 3 5 8 amma@amma05:~/Documents$
```

3.Selection Sort

Code:

```c
//selection
#include <stdio.h>
int main() {
    int a[] = {5, 2, 8, 1, 3};
    int n = 5;
    int i, j, min, temp;

    for(i = 0; i < n - 1; i++) {
        min = i;
        for(j = i + 1; j < n; j++) {
            if(a[j] < a[min])
                min = j;
        }
        temp = a[i];
        a[i] = a[min];
        a[min] = temp;
    }

    printf("Selection Sorted: ");
    for(i = 0; i < n; i++)
        printf("%d ", a[i]);

    return 0;
}
```

Output:

```
Insertion Sorted:
1 2 3 5 8 amma@amma05:~/Documents$ gcc selection.c -o selection
amma@amma05:~/Documents$ ./selection
Selection Sorted: 1 2 3 5 8 amma@amma05:~/Documents$
```

4.Heap Sort:

Code:

```c
//heap sort
#include <stdio.h>
void swap(int* a, int* b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
void heapify(int arr[], int n, int i) {
int largest = i;
int left = 2 * i + 1;
int right = 2 * i + 2;
if (left < n && arr[left] > arr[largest])
largest = left;
if (right < n && arr[right] > arr[largest])
        largest = right;
if (largest != i) {
        swap(&arr[i], &arr[largest]);
        heapify(arr, n, largest);
    }
}
void heapSort(int arr[], int n) {
for (int i = n / 2 - 1; i >= 0; i--) {
        heapify(arr, n, i);
    }
```

```c
void heapSort(int arr[], int n) {
for (int i = n / 2 - 1; i >= 0; i--) {
        heapify(arr, n, i);
    }
for (int i = n - 1; i > 0; i--) {
swap(&arr[0], &arr[i]);
heapify(arr, i, 0);
    }
}
void printArray(int arr[], int n) {
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    printf("\n");
}
int main() {
 int arr[] = {12, 11, 13, 5, 6, 7};
int n = sizeof(arr) / sizeof(arr[0]);
printf("Unsorted array: \n");
printArray(arr, n);
heapSort(arr, n);
printf("Sorted array: \n");
    printArray(arr, n);

    return 0;
}
```

Output:

```
jyothirmai@jyothirmai:~/Documents/main$ gcc heap.c -o heap
jyothirmai@jyothirmai:~/Documents/main$ ./heap
Unsorted array:
12 11 13 5 6 7
Sorted array:
5 6 7 11 12 13
jyothirmai@jyothirmai:~/Documents/main$
```