EXERCISE 19

Write a C program to implement Heap sort

Aim:

To write a C program to sort a series of numbers using the Heap Sort algorithm.

Algorithm:

1. Build a Max Heap from the input data.
2. Swap the root (maximum value) with the last item.
3. Reduce the heap size by one and heapify the root.
4. Repeat the process until the heap size is 1.

Program:

```
#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;      // Initialize largest as root
   int left = 2 * i + 1; // left child
   int right = 2 * i + 2; // right child
   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]);
   }
```

```
}
void heapSort(int arr[], int n) {
    // Build max heap
    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 display(int arr[], int size) {
    printf("Sorted array:\n");
    for (int i = 0; i < size; i++)
        printf("%d ", arr[i]);
    printf("\n");
}
int main() {
    int arr[50], n;
    printf("Enter number of elements: ");
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++)
        scanf("%d", &arr[i]);
    heapSort(arr, n);
    display(arr, n);
    return 0;
```

}

Input and Output:

```
Enter number of elements: 4
Enter 4 elements:
33 45 67 1
Sorted array:
1 33 45 67


=== Code Execution Successful ===
```

Result :

The series of numbers has been successfully sorted using the Heap Sort method**.**