

Heap Sort

```
#include <stdio.h>

void heapify(int arr[], int i, int size)
{
    int left = 2 * i + 1;
    int right = 2 * i + 2;
    int maxIdx = i;

    if (left < size && arr[left] > arr[maxIdx])
    {
        maxIdx = left;
    }
    if (right < size && arr[right] > arr[maxIdx])
    {
        maxIdx = right;
    }

    if (maxIdx != i)
    {
        int temp = arr[maxIdx];
        arr[maxIdx] = arr[i];
        arr[i] = temp;
        heapify(arr, maxIdx, size);
    }
}

void heapSort(int arr[], int n)
{
    int i;
    for (i = n / 2; i >= 0; i--)
    {
        heapify(arr, i, n);
    }
}
```

```

    }

    for (i = n - 1; i >= 0; i--)
    {
        int temp = arr[0];
        arr[0] = arr[i];
        arr[i] = temp;
        heapify(arr, 0, i);
    }
}

int main()
{
    int n, i;
    printf("\nEnter the number of elements: ");
    scanf("%d", &n);

    int arr[n];

    printf("Enter array elements: ");
    for (i = 0; i < n; i++)
    {
        scanf("%d", &arr[i]);
    }

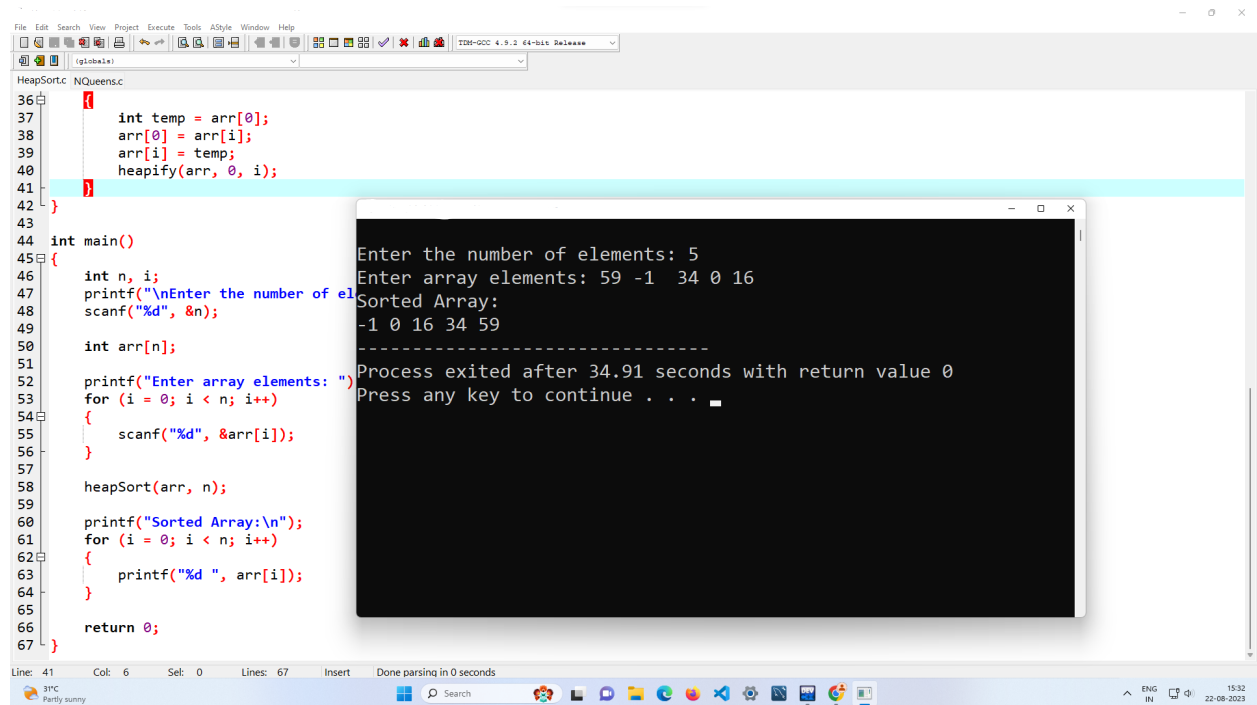
    heapSort(arr, n);

    printf("Sorted Array:\n");
    for (i = 0; i < n; i++)
    {
        printf("%d ", arr[i]);
    }

    return 0;
}

```

OUTPUT:



The screenshot displays a C++ IDE with a file named 'HeapSort.cpp' and a terminal window showing the program's execution. The code implements a heap sort algorithm. The terminal output shows the user entering 5 elements, followed by the array elements 59, -1, 34, 0, and 16. The sorted array is then displayed as -1, 0, 16, 34, and 59. The program exits after 34.91 seconds with a return value of 0.

```
36  
37     int temp = arr[0];  
38     arr[0] = arr[i];  
39     arr[i] = temp;  
40     heapify(arr, 0, i);  
41  
42 }  
43  
44 int main()  
45 {  
46     int n, i;  
47     printf("\nEnter the number of elements: ");  
48     scanf("%d", &n);  
49  
50     int arr[n];  
51  
52     printf("Enter array elements: ");  
53     for (i = 0; i < n; i++)  
54     {  
55         scanf("%d", &arr[i]);  
56     }  
57  
58     heapSort(arr, n);  
59  
60     printf("Sorted Array:\n");  
61     for (i = 0; i < n; i++)  
62     {  
63         printf("%d ", arr[i]);  
64     }  
65  
66     return 0;  
67 }
```

```
Enter the number of elements: 5  
Enter array elements: 59 -1 34 0 16  
Sorted Array:  
-1 0 16 34 59  
-----  
Process exited after 34.91 seconds with return value 0  
Press any key to continue . . .
```