```
#include <stdio.h>
#include <stdlib.h>
int heap_size = 6;
int size = 7;
void swap(int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}
void max_heap(int a[], int i)
{
    int l = 2i + 1;
    int r = 2i + 2;
    int large = i;
    if (l < size && a[l] > a[i])
    {
        large = l;
    }
    if (r < size && a[r] > a[large])
    {
        large = r;
    }
    if (large != i)
    {
        swap(&a[i], &a[large]);
        max_heap(a, large);
    }
}

void insertkey(int a[], int i, int key)
{
    if (100 <= heap_size)
    {
        printf("Overflow: Could not insertKey\n");
        return;
    }

    heap_size++;
    i = heap_size - 1;
    a[i] = key;

    while (i != 0 && a[i] < a[i / 2])
    {
        swap(&a[i], &a[i / 2]);
        i = i / 2;
    }
}
int extract_max(int a[], int size)
{
    if (size <= 0)
    {
        printf("heap undefined\n");
    }
    /*if (heap_size == 1)
    {
        heap_size--;
        return a[0];
    }*/

    int max = a[0];

    a[0] = a[size - 1];
    //heap_size--;
    max_heap(a, size);

    return max;
}

int main()
{
    int a[100];
    a[0] = 1;
    a[1] = 3;
    a[2] = 6;
    a[3] = 5;
    a[4] = 9;
    a[5] = 8;
    int k[100];
    printf("Enter elements to insert\n");
    int n;
    scanf("%d", &n);
    for (int j = 0; j < n; j++)
    {
        scanf("%d", &k[j]);
        insertkey(a, j, k[j]);
    }
    int size = sizeof(a) / sizeof(a[0]);
    for (int i = 0; i < heap_size; i++)
    {
        printf("%d ", a[i]);
    }

    /* int a[] = {25, 14, 16, 13, 10, 8, 12};
    int size = sizeof(a) / sizeof(a[0]);
    printf("hi");
    max_heap(a, 0);

    for (int j = 0; j < size; j++)
    {
        int b = extract_max(a, size - j);
        printf("%d ", b);
    }*/
}
```