3. Min and Max Heap using arrays

Description: Build min-heap and max-heap, insert, delete-root operations, and display.

```c
// heap.c
#include <stdio.h>
#include <stdlib.h>

void swap(int *a,int *b){int t=*a;*a=*b;*b=t;}

// Max-heapify
void maxHeapify(int arr[], int n, int i){
    int largest = i; int l = 2*i+1; int r = 2*i+2;
    if(l<n && arr[l]>arr[largest]) largest=l;
    if(r<n && arr[r]>arr[largest]) largest=r;
    if(largest!=i){ swap(&arr[i], &arr[largest]); maxHeapify(arr,n,largest);}
}

// Min-heapify
void minHeapify(int arr[], int n, int i){
    int smallest=i; int l=2*i+1; int r=2*i+2;
    if(l<n && arr[l]<arr[smallest]) smallest=l;
    if(r<n && arr[r]<arr[smallest]) smallest=r;
    if(smallest!=i){ swap(&arr[i], &arr[smallest]);
minHeapify(arr,n,smallest);} }

void buildMaxHeap(int arr[], int n){ for(int i=n/2-1;i>=0;i--)
maxHeapify(arr,n,i); }
void buildMinHeap(int arr[], int n){ for(int i=n/2-1;i>=0;i--)
minHeapify(arr,n,i); }

int main(){
    int n; printf("Enter number of elements: "); scanf("%d", &n);
    int *arr = (int*)malloc(sizeof(int)*n);
    for(int i=0;i<n;i++) scanf("%d", &arr[i]);
    int choice; printf("1: Max-Heap 2: Min-Heap\nChoose: "); scanf("%d",
&choice);
    if(choice==1){ buildMaxHeap(arr,n); printf("Max-Heap array: "); for(int
i=0;i<n;i++) printf("%d ", arr[i]); }
    else { buildMinHeap(arr,n); printf("Min-Heap array: "); for(int
i=0;i<n;i++) printf("%d ", arr[i]); }
    printf("\n"); free(arr); return 0;
}
```