Week-8 UE20CS207 DSLAB

- Name : P K Navin Shrinivas
- SRN : PES2UG20CS237
- Section : D
- Batch : 2

Assginment problem 1 : Max heapy , heapify using bottom up approach

Code :

main.c

```c
#include "1_1.h"
#include <stdio.h>



// n/2 will be the posistion for the node for the last child!!!!!
// log : n is the last child , n/2 is ofc its nide


int main(){
    int tree[100]={0};
    while(true){
        printf("1.Insert \n");
        int choice;
        printf("Enter choice :");
        scanf("%d" , &choice);
        if(choice == 1){
            int d;
            printf("Enter element to insert : ");
            scanf("%d",&d);
            HeapInsert(tree , d);
        }
    }
}
```

1_1.h

```c

#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#define ARRSIZE 1000

void HeapInsert(int* tree , int d);
```

5_1.c

```c
#include "1_1.h"
#include <stdio.h>


void HeapifyMax(int* tree ,int i, int n){
    int left = 2*i;
    int right = (i*2)+1;
    int largest = i;
    if(left<=n && tree[left]>tree[largest])
        largest = left;

    if(right<=n && tree[right]>tree[largest])
        largest = right;

    if(largest != i)
    {
        int temp = tree[largest];
        tree[largest]=tree[i];
        tree[i] = temp;
        HeapifyMax(tree , largest , n);
    }

}


void HeapInsert(int* tree , int d){
    int n=0;
    for(int i=1;i<ARRSIZE ; i++){
        if(tree[i] == 0){
            n=i;
            tree[n]=d;
        }
        else
            continue;
    }
    for(int i=n/2;i>=1 ;i--){
        HeapifyMax(tree, i , n);
    }
    printf("\nInserted Element to Heap! \n");
}```
```