1.结构体：

存值：堆数据类型 data

左子树：结构体\* left

右子树：结构体\* right

错误：堆用数组实现，所以是size和capacity

2.函数：保持接口统一：传一级指针

断言：

Assert

If

typedef int HPDataType;

typedef struct Heap

{

HPDataType\* a;

int size;

int capacity;

}HP;

//断言

void assert\_pointer(void\* pointer);

//初始化

void HeapInit(HP\* php);

//建堆

void HeapInitArray(HP\* php, int\* a, int n);

//销毁

void HeapDestroy(HP\* php);

//添加

void HeapPush(HP\* php, HPDataType x);

//删除

void HeapPop(HP\* php);

//查值

HPDataType HeapTop(HP\* php);

//判空

bool HeapEmpty(HP\* php);

//查大小

int HeapSize(HP\* php);

//向上调整

void AdjustUp(HPDataType\* a, int child);

//向下调整

void AdjustDown(HPDataType\* a, int n, int parent);

//交换

void Swap(HPDataType\* p1, HPDataType\* p2);