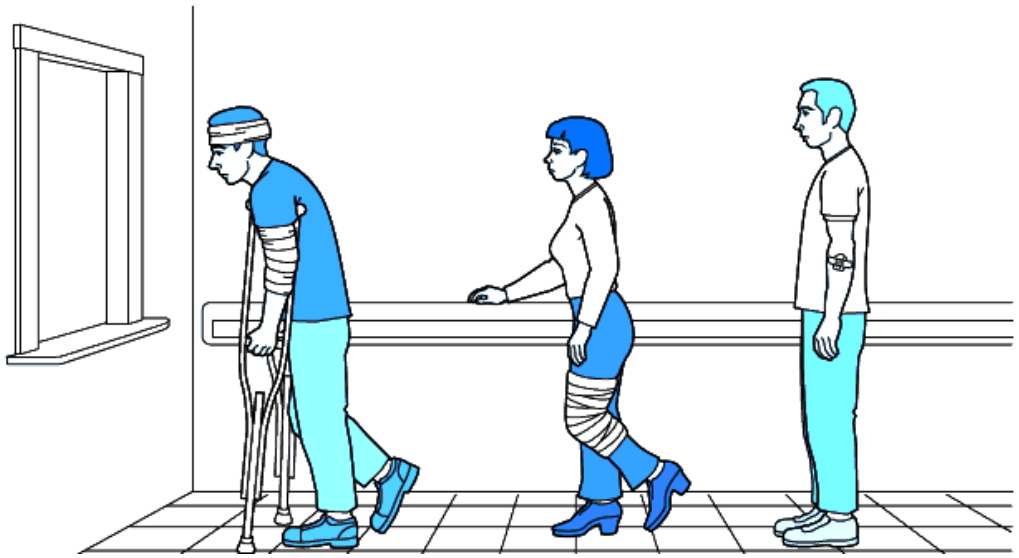
Priority Queue

(ถอยคอยลำดับความสำคัญ, แถวคอยเชิงบุริมภาพ)

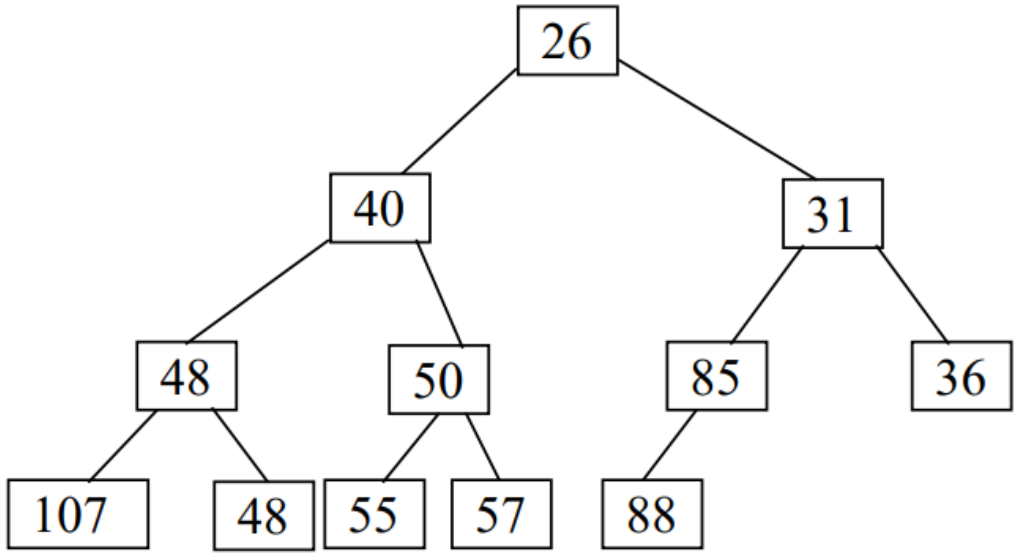
By Payongkit XI

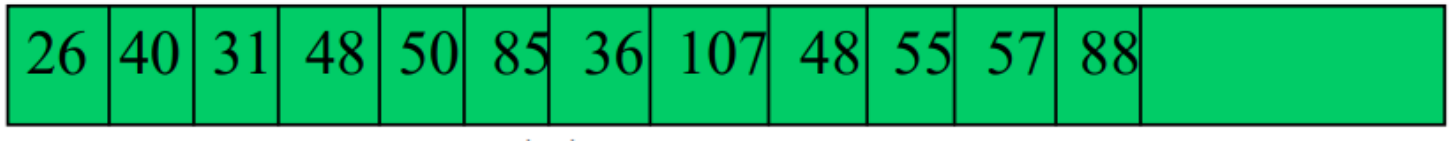
Priority Queue

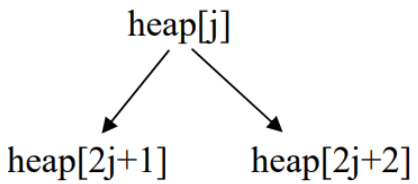
(ถอยคอยลำดับความสำคัญ, แถวคอยเชิงบุริมภาพ)



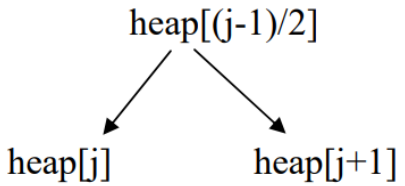
Heap





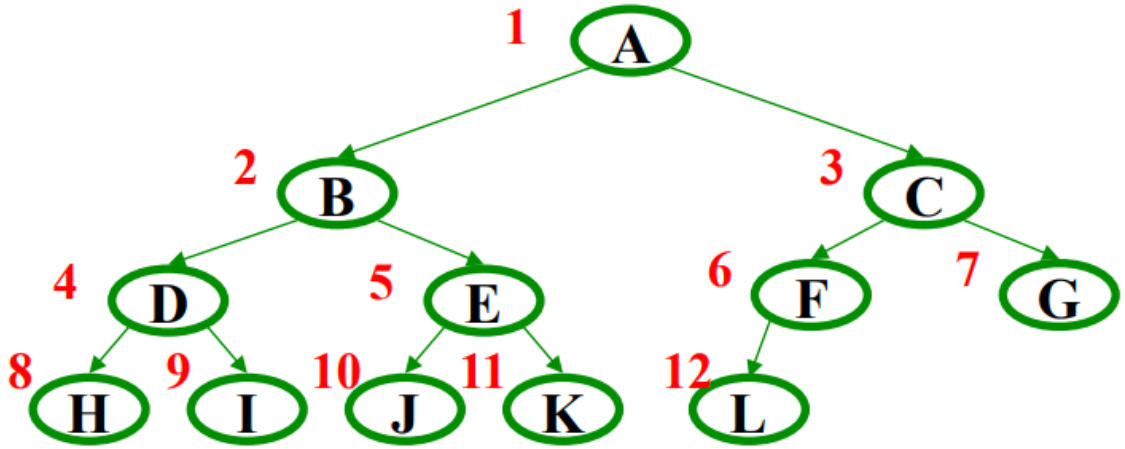


ค่าดัชนีของโนดลูก ทั้งซ้ายและขวา
นับลงมาจากโนด parent



ค่าดัชนีของโนด parent
เมื่อนับขึ้นไปจากโนดลูก

Array Representation



From node `i`:

left child: `i*2`
right child: `i*2+1`
parent: `i/2`

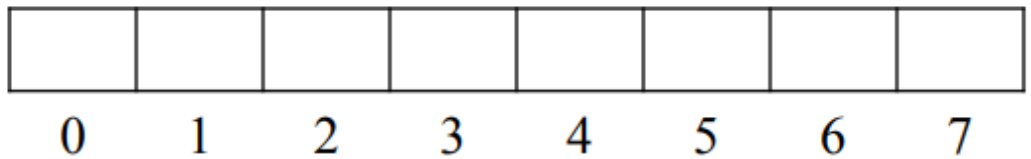
(wasting index 0 is convenient for the index arithmetic)

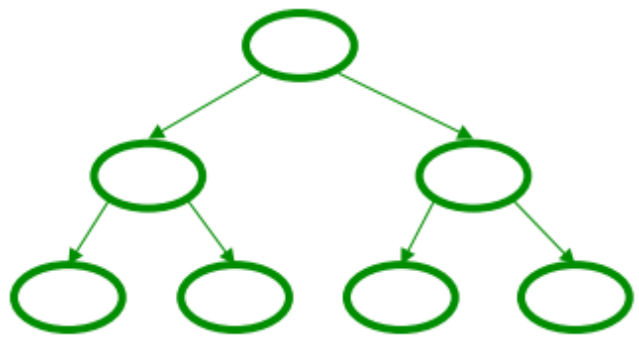
implicit (array) implementation:

	A	B	C	D	E	F	G	H	I	J	K	L	
0	1	2	3	4	5	6	7	8	9	10	11	12	13

Example

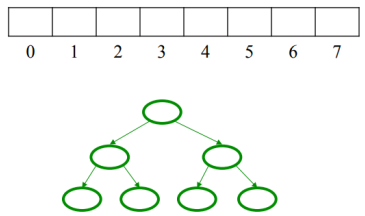
1. insert: 16, 32, 4, 67, 105, 43, 2
2. deleteMin

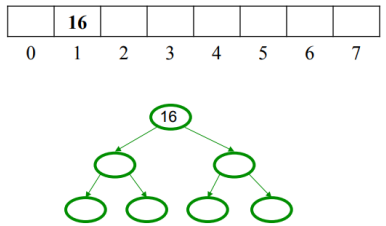


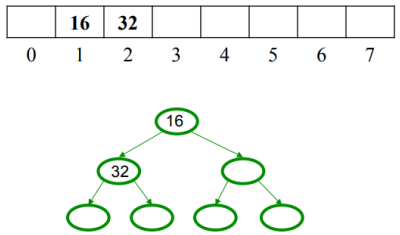


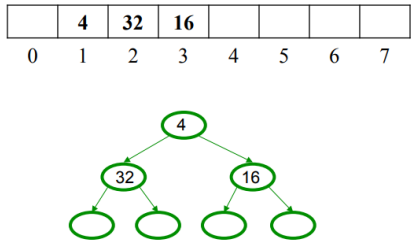
Example

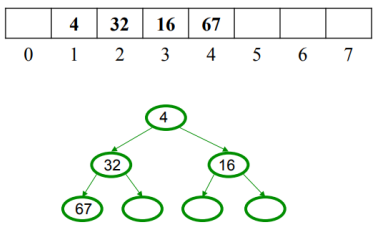
1. insert: 16, 32, 4, 67, 105, 43, 2

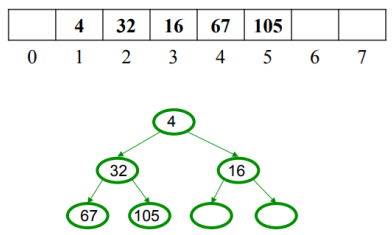


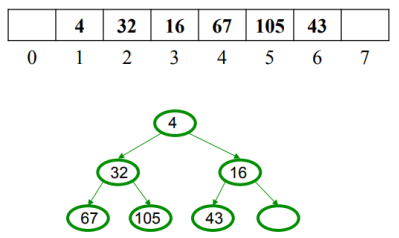


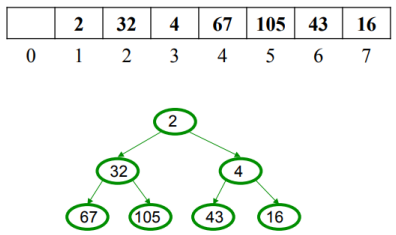




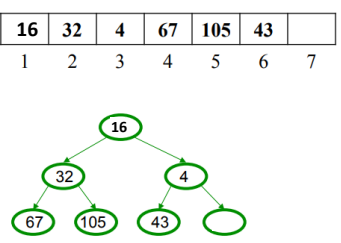


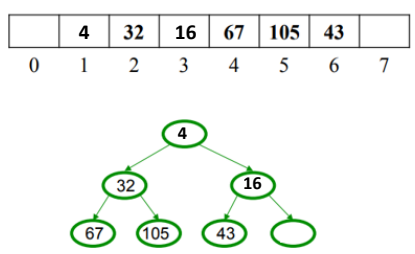






2. deleteMin





Pseudocode: insert

```
void insert(int val) {
  if(size==arr.length-1)
    resize();
  size++;
  i=percolateUp(size,val);
  arr[i] = val;
}
```

```
int percolateUp(int hole,
                int val) {
  while(hole > 1 &&
        val < arr[hole/2])
    arr[hole] = arr[hole/2];
    hole = hole / 2;
  }
  return hole;
}
```

Pseudocode: deleteMin

```
int deleteMin() {
  if(isEmpty()) throw…
  ans = arr[1];
  hole = percolateDown
           (1,arr[size]);
  arr[hole] = arr[size];
  size--;
  return ans;
}
```

```
int percolateDown(int hole,
                  int val) {
 while(2*hole <= size) {
  left  = 2*hole;
  right = left + 1;
  if(right > size ||
     arr[left] < arr[right])
    target = left;
  else
    target = right;
  if(arr[target] < val) {
    arr[hole] = arr[target];
    hole = target;
  } else
      break;
 }
 return hole;
}
```