5주차 실습과제

20135151 이갑성

1번

//=======Heap========

**class** Heap1 {

**int** maxSize; // Heap의 최대 크기

**int** n; // 현재 Heap크기

**int** heap[]; // Heap tree안에 있는 노드 저장 배열

Heap1() {

maxSize = 50;

n = 0;

heap = **new** **int**[maxSize]; // size 50인 배열 생성

}

**public** **void** insertHeap(**int** item)

{

**if**(n == 0)

{

n++;

heap[n] = item;

}

**else**

{

**int** temp;

n++;

heap[n] = item; //1.값 넣기

//2.아버지와 자식을 비교하기

**if**(heap[n] > heap[n/2])

{

**for**(**int** i = n; i > 1; i = i / 2) //i = i / 2를 해줌으로써 아버지하고만 비교하게 된다. // i>1인 것은 i > 1까지 여야 i/2를 하여 아버지와 비교할수있다.

{

**if**(heap[i] > heap[i/2])

{

temp = heap[i];

heap[i] = heap[i/2];

heap[i/2] = temp;

}

**else**

**break**;

}

}

}

}

**public** **int** deleteHeap()

{

**int** item, temp;

**int** i, j;

**if**(n == 0)

{

**return** 0;

}

**else**

{

item = heap[1];

temp = heap[n];

n = n - 1;

i = 1;

j = 2;

**while**(i <= n)

{

**if**(j < n)

{

**if**(heap[j] < heap[j+1])

{

j = j + 1;

}

}

**if**(temp >= heap[j])

**break**;

heap[i] = heap[j];

i = j;

j = j \* 2;

}

heap[i] = temp;

**return** item;

}

}

**public** **void** printHeap() {

**for** (**int** i = 1; i <= n; i++) {

System.***out***.print(heap[i] + " ");

}

System.***out***.println();

}// end printHeap()

}

// =======메인=======

**public** **class** HeapTest1 {

**public** **static** **void** main(String[] args) {

Heap1 hp = **new** Heap1();

hp.insertHeap(18);

hp.insertHeap(13);

hp.insertHeap(5);

hp.insertHeap(12);

hp.insertHeap(8);

hp.insertHeap(19);

System.***out***.print("입력 된 값 : ");

hp.printHeap();

System.***out***.println("");

**boolean** a = **true**;

**int** ob;

**while** (a) {

ob = hp.deleteHeap();

**if** (ob != 0) {

System.***out***.println("삭제 된 값 : " + ob);

System.***out***.print("삭제 후 값 : ");

hp.printHeap();

System.***out***.println("");

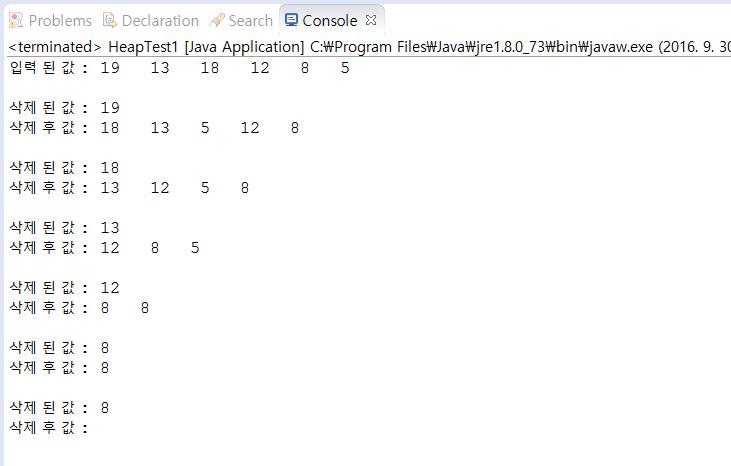
} **else**

a = **false**;

}

}

}



2번

**class** Heap2 {

**int** maxSize;

**int** heap[];

**int** n;

Heap2() {

maxSize = 50;

heap = **new** **int**[maxSize];

n = 0;

}

**void** makeTreeHeap(**int** H[], **int** cnt)

{

**int** p, temp;

**for**(**int** i = cnt / 2; i >= 1; i--)

{

p = i;

**for**(**int** j = 2\*p; j <= cnt; j = 2\*j)

{

**if**(j < cnt)

{

**if**(H[j] < H[j+1])

{

j = j + 1;

}

}

**if**(H[p] >= H[j])

**break**;

temp = H[p];

H[p] = H[j];

H[j] = temp;

p = j;

}

}

} // end makeTreeHeap()

}

**public** **class** HeapTest2 {

**public** **static** **void** main(String[] args) {

Heap2 hp2 = **new** Heap2();

**int** a[] = { 0, 15, 11, 70, 5, 13, 50}; // 완전이진트리구현 0은 제외될 값

System.***out***.println("완전이진트리의 값");

**for** (**int** i = 1; i < a.length; i++) {

System.***out***.print(a[i] + " ");

}

System.***out***.println("\n");

hp2.makeTreeHeap(a, a.length - 1); // 배열과 배열의 크기를 넘겨준다.

System.***out***.println("Max\_Heap의 값");

**for** (**int** i = 1; i < a.length; i++) {

System.***out***.print(a[i] + " ");

}

}

}

