

자료 구조 숙제 # 4

제출일 : 2019년 6월 9일 일요일 (eCampus)

제출물 : 학번.java 파일 하나

Hard copy 제출 필요 없음

전체 구현 30점

Hw194.zip 파일 : LabTest.java hw04.java lab.in lab.out

이번 숙제는 Heap Sort 알고리즘을 구현하는 내용이다. Hw04.java에는 HeapSort라는 클래스가 구현되어 있다. 사용자로부터 숫자를 입력 받고, 이 숫자들을 heap sort 과정을 이용해서 정렬을 하는데, 그 알고리즘의 각 단계마다 내부적으로 max heap을 위해 사용하는 배열의 내용을 출력시킨다.

수행 예는 다음과 같다.

```
sanghwan@sanghwan-VirtualBox: ~/dbox/classes191/ds/hw/hw194
File Edit View Search Terminal Help
sanghwan@sanghwan-VirtualBox:~/dbox/classes191/ds/hw/hw194$ java LabTest
HeapSort > init 12 17 14 1 21 -1
Heap : - 12 17 14 1 21
HeapSort > sort
Heap : - 21 17 14 1 12
Heap : - 17 12 14 1 21
Heap : - 14 12 1 17 21
Heap : - 12 1 14 17 21
Heap : - 1 12 14 17 21
Heap : - 1 12 14 17 21
HeapSort > init 1 2 3 4 -1
Heap : - 1 2 3 4
HeapSort > sort
Heap : - 4 2 3 1
Heap : - 3 2 1 4
Heap : - 2 1 3 4
Heap : - 1 2 3 4
Heap : - 1 2 3 4
HeapSort > quit
sanghwan@sanghwan-VirtualBox:~/dbox/classes191/ds/hw/hw194$
```

사용자가 사용하는 명령어의 syntax는 다음과 같다. main() 함수에 정의되어 있다.

- init heapvalues

정렬할 숫자들을 입력받는 과정을 수행한다. 주어진 정수 값들을 -1이 나올 때까지 읽

어서 그 내용대로 heap의 1번 index부터 차례로 집어 넣는다. 이미 다 구현되었으니 그냥 사용하기만 하면 된다. 숫자들은 임의의 순서로 들어온다.

- `sort`

사용자로부터 입력 받은 숫자를 오름차순으로 정렬한다.

클래스 `HeapSort`에는 `heap`이라는 멤버 변수가 있는데 이 변수는 `array` 타입으로 사용자로부터 입력 받은 숫자를 저장한다. 그 숫자의 개수는 `heapSize`에 저장되어 있고, 숫자들은 배열의 인덱스 1부터 `heapSize`까지 차례대로 저장된다. 배열 인덱스 0은 사용하지 않는다. `Heap Sort`를 구현하기 위해 다음 2 함수를 구현해야 한다.

- `void Adjust(int root, int n);`

인덱스 `root`에서부터 시작하는 subtree를 max heap 상태를 만족하도록 변경한다. 이 트리의 맨 마지막 노드는 index `n`에 저장되어 있다. 이때 `root`를 제외한 나머지는 모두 heap 상태를 만족하고 있다고 가정하면 된다.

- `void Sort();`

우선 heap 배열을 max heap의 상태로 변경한다. 즉 Initialization을 수행한다. 이를 위해 위에서 구현한 `Adjust` 함수를 적절하게 호출해야 한다. Initialization이 끝난 후 `System.out.println(this);` 문장을 실행하여 heap 배열의 상태를 출력한다. 그 후에 max heap 상태의 배열을 이용하여 정렬을 수행한다. 알고리즘은 책이나 강의 자료를 참조한다.

최대값을 하나씩 heap 배열의 뒷부분으로 이동할 때마다 그 현재 heap 배열의 내용을 출력한다. 이를 위해서는 `System.out.println(this);` 문장을 실행하면 된다.

프로그램 테스트

```
$ diff abc lab.out
```

또는

```
$ diff -i --strip-trailing-cr -w abc lab.out
```