

자료 구조 Lab 004 :

lab004.zip 파일 : LabTest.java lab004.java lab.in lab.out

제출

lab004.java 를 lab4_학번.java 로 변경하여 이 파일 한 개만 제출할 것.

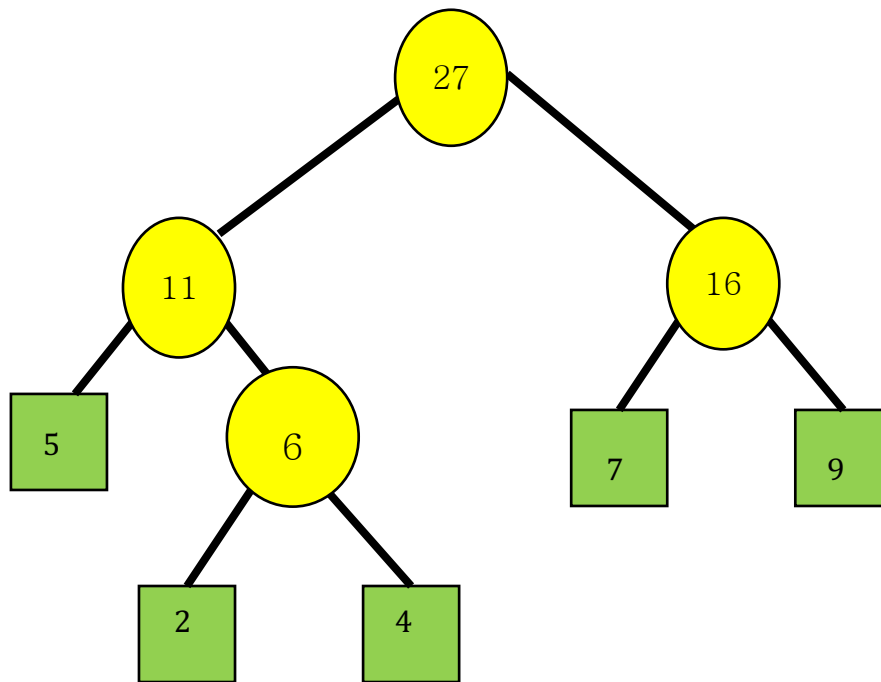
다음은 Huffman 알고리즘을 구현하는 과제이다.

수행 예는 다음과 같다.

```
sanghwan@Sanghwanui-MacBookPro lab24004 % javac LabTest.java lab004sol.java
sanghwan@Sanghwanui-MacBookPro lab24004 % java LabTest
Huff > huff 2 5 4 7 9 -1
Runs : 2 5 4 7 9
Merging 2 4
Merging 5 6
Merging 7 9
Merging 11 16
Huff > huff 14 3 6 9 -1
Runs : 14 3 6 9
Merging 3 6
Merging 9 9
Merging 14 18
Huff > huff 1 2 3 4 -1
Runs : 1 2 3 4
Merging 1 2
Merging 3 3
Merging 4 6
Huff > █
```

런의 크기를 의미하는 숫자를 입력 받은 다음 런의 크기를 출력하고, 런들을 Huffman 알고리즘으로 머지하는 과정을 보여준다.

위 예에서 첫번째 huff 명령은 아래와 같은 Merge 과정을 보여준다.



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

- `huff numbers`

사용자가 숫자를 입력하고 맨 마지막 숫자는 -1로서 입력의 끝을 표시한다. 이 숫자들을 Merge하는 과정을 출력한다.

이 내용을 구현하기 위해 다음 하나의 함수를 구현해야 한다.

- `void MergeRuns();`

`class Huffman`에는 다음 2개의 변수가 있다.

```
int noe; // the number of elements
private int[] inputArray; // input array
```

이중 `noe`는 원소의 개수를 의미하고, `inputArray`는 입력된 숫자들이 들어 있는 array를 의미한다. 우선순위 큐를 이용하여 이 알고리즘을 구현할 수 있는데, `java.util`에서 제공하는 `PriorityQueue` 클래스를 활용한다.

```
PriorityQueue<Integer> pq = new PriorityQueue<Integer>();
```

<https://docs.oracle.com/javase/8/docs/api/java/util/PriorityQueue.html>

프로그램 결과 테스트

컴파일

```
$ javac lab004.java LabTest.java
```

실행

```
$ java LabTest
```

주어진 **input**으로 실행

```
$ java LabTest < lab.in
```

주어진 **output**과 비교

```
$ java LabTest < lab.in > abc
```

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