자료 구조 Lab 008 :

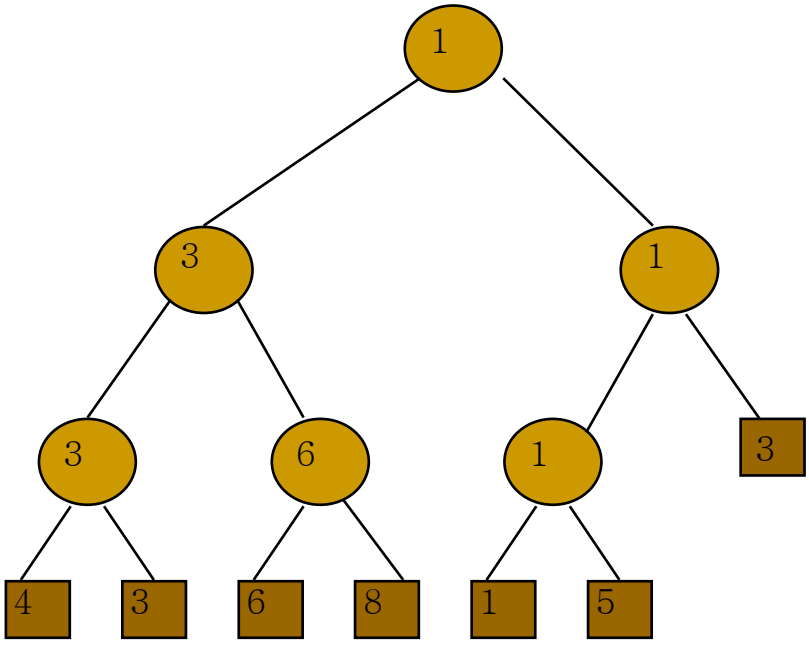
lab008.zip 파일 : LabTest.java lab008.java lab.in lab.out lab008.pdf

---

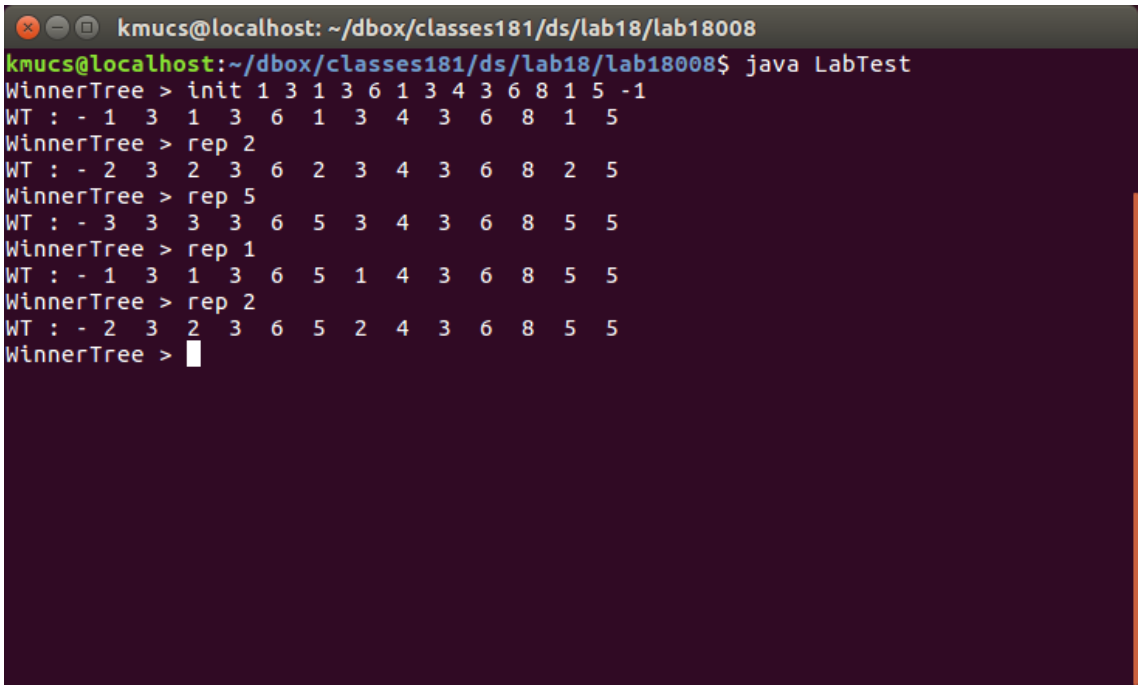
제출

lab008.java 를 학번.java 로 변경하여 이 파일 한 개만 제출할 것.

다음은 배열을 이용하여 Winner Tree를 구현하는 내용이다. 클래스 WinnerTree는 아래 그림과 같이 complete binary tree의 형태를 가지고 있는데, complete binary tree이기 때문에 배열로 표현하는 것이 적절하다.



이 프로그램에서는 사용자로부터 위와 같은 Winner Tree의 내용을 입력 받은 후, winner 원소를 사용자의 rep 명령에 의해 제거하고, 이 rep 명령에 의해 주어진 값으로 winner 원소의 external node가 치환되고 winner tree가 재구성한다. 수행 예는 다음과 같다.



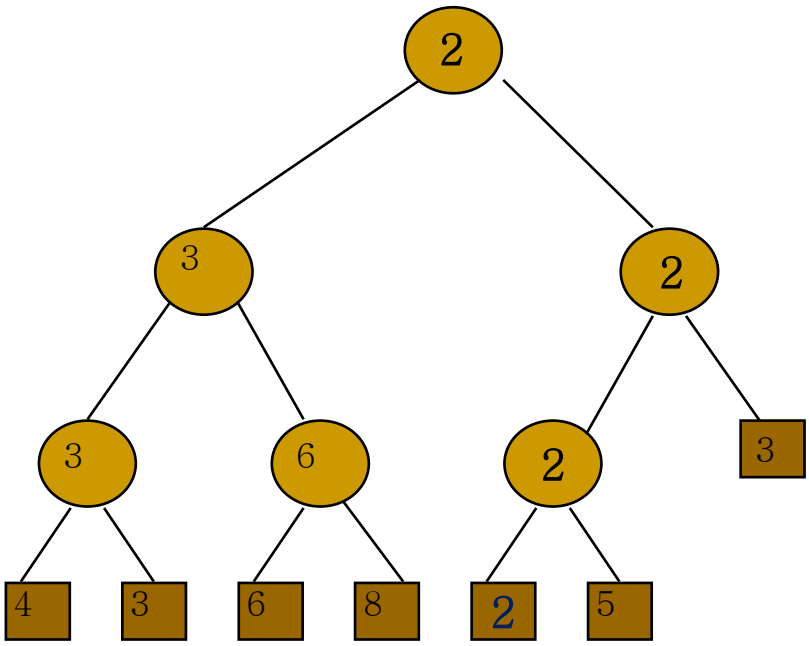
사용자가 사용하는 명령어의 syntax는 다음과 같다. Main() 함수에 정의되어 있다.

- init values

  init 명령어로 Winner Tree를 구성하는 internal node와 external node의 값를 입력 받은 다음, 이를 바탕으로 Winner Tree를 생성하여 그 결과를 보여주는데 현재 구현 방식이 배열을 이용하는 것이기 때문에 그냥 배열의 상태를 보여준다. 위 실행 예의 첫 번째 init은 앞의 그림의 Winner Tree가 배열에 저장된 상태를 보여준다. Index 0가 사용되지 않기 때문에 -로 표현했다. 이미 구현되어 있다.

- rep value

  이 명령에 의해 주어진 value값으로 winner의 external node를 치환하고, 트리를 재구성한다. init 명령이 새로 불리기 전까지는 기존의 winner tree에 계속 적용한다. 이 명령이 수행된 후에도 배열의 상태를 보여준다. 다음 그림은 위 예제에서 첫 번째 "rep 2"에 의해 변화된 상태를 보여준다.



이 내용을 구현하기 위해 다음 메소드를 구현해야 한다.

- `void PopNReplace(int data);`

  Root에 해당하는 External node를 찾아서 그 값을 주어진 data로 치환하고, 트리를 재구성한다. Root에 해당하는 External node가 여러 개일 경우 가장 왼쪽에 있는 노드를 사용한다. 필요시 WinnerTree 클래스 내부에 새로운 메소드를 추가적으로 구현하여도 된다.

주어진 .java 파일을 컴파일 하면 수행은 가능하지만 아직 구현이 안된 부분은 "NEED TO IMPLEMENT" 라는 comment가 존재한다.

프로그램 결과 테스트

```
$ diff aaa lab.out
```

또는

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