2019학년도 2학기

JAVA 프로그래밍 II

514770
2019년 가을학기
10/8/2019
박경신

과제 Lab5_1 (ArrayList)

1. Lab5_1은 Lab4의 PersonManager에 iterator를 구현한다.

```
public class PersonManager implements Iterable<Person> {
    private List<Person> data = null;
    public Iterator<Person> iterator() {
        return new PersonManagerIterator();
    } ....
}
```

2. PersonManager 안에 **PersonManagerIterator** 내부 클래스(inner class)를 구현한다.

```
private class PersonManagerIterator implements Iterator<Person> {
    public boolean hasNext() { ... } // 내부 구현 요망
    public Person next() { ... } // 내부 구현 요망
    public void remove() { .... } // 내부 구현 요망
}
```

과제 Lab5_1 (Dynamic array)

1. public class PersonArrayManager implements Iterable<Person>의 내부 구현을 **Dynamic Array (ArrayList 같은)** 방식으로 구현한다.
    1. private Person[] data = null;
    2. private int count = 0;
    3. private int size = 0;
2. PersonArrayManager 안에 **PersonArrayManagerIterator** 내부 클래스(inner class)를 구현한다.

```
private class PersonArrayManagerIterator implements Iterator<Person> {
    public boolean hasNext() { ... } // 내부 구현 요망
    public Person next() { ... } // 내부 구현 요망
    public void remove() { .... } // 내부 구현 요망
}
```

3. **PersonManager 와 PersonArrayManager 클래스 메소드를 테스트한다. 그리고, ArrayList와 Dynamic Array 방식을 비교한다.**

과제 Lab5_1 (==, equals, hashCode, Collection)

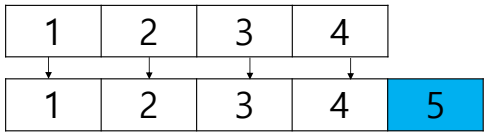
1. Person, BMICalculator 클래스에 equals과 hashCode를 override하라.
2. PersonManager 클래스에 다음 메소드를 추가한다.
    1. public boolean contains(T data);
    2. public void sort(); // sort by Comparable
    3. public void sort(Comparator<? super T> comparator); // sort by Comparator
3. **==, equals, hashCode를 테스트하는 프로그램을 작성한다. 그리고, Vector, ArrayList, LinkedList, HashSet, HashMap을 사용하여 테스트하고 자료구조를 비교 분석하라.**

과제 제출

- Lab4_2 예제를 사용하여, 그것을 관리하는 Manager 클래스를 본인이 원하는 자료 구조형 (예시: DoublyLinkedList)을 사용하여 작성하라.
- Lab5_1 ~ Lab5_2와 보고서를 전체적으로 묶어서 e-learning에 과제 제출

Dynamic Array

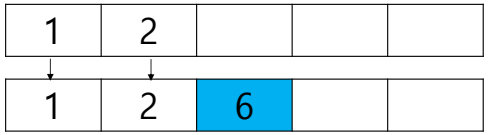
- Dynamic Array
  - Add an element at the end of an array



count=4
size=4

count++
size++

  - Add an element at the end of an array

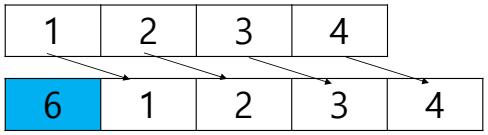


count=2
size=5

count++

Dynamic Array

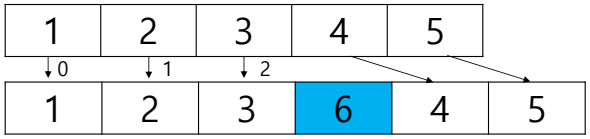
- Dynamic Array
  - Insert an element at the index 0



count=4
size=4

count++
size++

  - Insert an element at the index 3

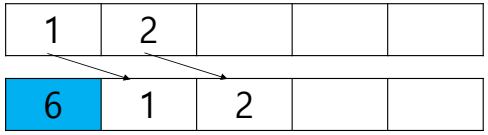


count=5
size=5

count++
size++

Dynamic Array

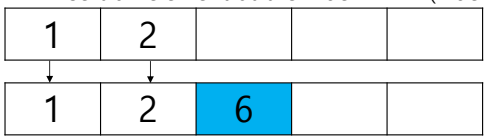
- Dynamic Array
  - Insert an element at the index 0



count=2
size=5

count++
size=5

  - Insert an element at the index 2 – if (index > count), ArrayIndexOutOfBoundsException

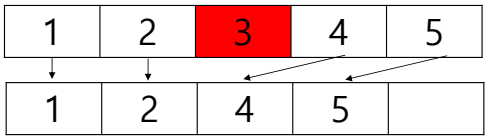


count=2
size=5

count++

Dynamic Array

- Dynamic Array
  - Remove an element at the index 2

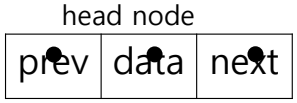


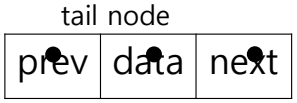
count=5
size=5

count--

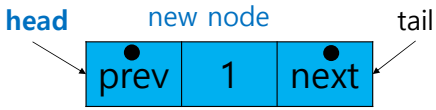
Doubly Linked List

- Doubly Linked List
  - Empty

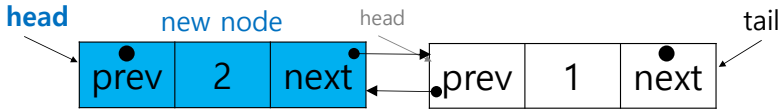




  - Add an element at the head

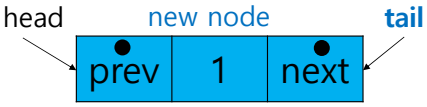


  - Add an element at the head

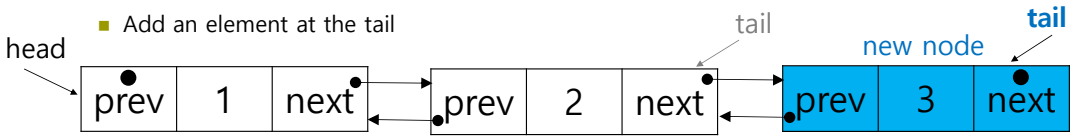


Doubly Linked List

- Doubly Linked List
  - Add an element at the tail

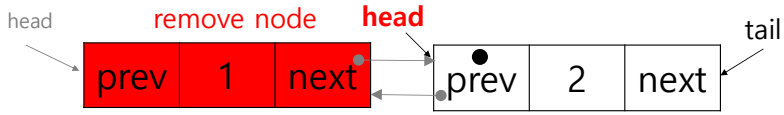


  - Add an element at the tail

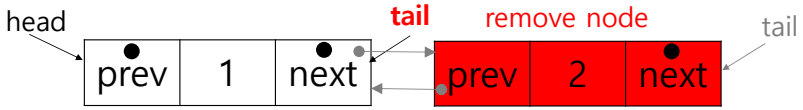


Doubly Linked List

- Doubly Linked List
  - Remove an element at the head

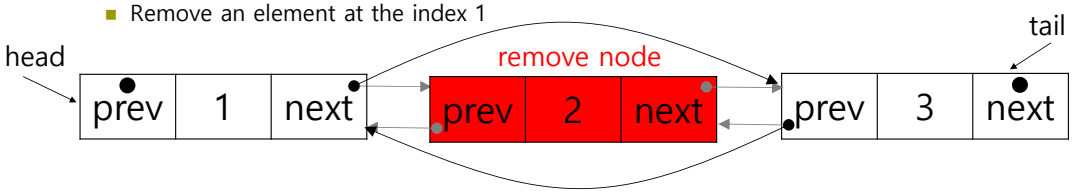


  - Remove an element at the tail



Doubly Linked List

- Doubly Linked List
  - Remove an element at the index 1



  - Insert an element at the index 1