

HW 10

Name: Zhenzhen Su

Date: 4/19/17

Instructor: Dr Charles Reid

Problem 1

Problem Statement: Create a method to count average vowels in an ArrayList of words

Code:

```
public class Main {
    public static void main(String[] args) {
        //count average vowels
        ArrayList<String> wordList = new ArrayList<String>();
        wordList.add("dsfadf");
        wordList.add("sd");
        wordList.add("dfafqvni");
        System.out.println("The average vowels of the list is " +
            averageVowels(wordList));
    }
}

public static double averageVowels(ArrayList<String> words) {
    ArrayList<Character> vowels = new ArrayList<Character>();
    vowels.add('a');
    vowels.add('e');
    vowels.add('i');
    vowels.add('o');
    vowels.add('u');

    int numOfVow = 0;
    for (String word : words) {
        word = word.toLowerCase();
        for (int i = 0; i < word.length(); i++) {
            if (vowels.contains(word.charAt(i))) {
                numOfVow++;
            }
        }
    }

    return numOfVow * 1.0 / words.size();
}
```

Console Output:

The average vowels of the list is 1.0

Problem 6

Problem statement: create minToFront method

Code:

```
public static void main(String[] args) {
    //minToFront method
    ArrayList<Integer> list = new ArrayList<Integer>();
    list.add(2);
    list.add(45);
    list.add(1);
    list.add(3);
    System.out.println("The intial list is " + list);
    System.out.println("The list after minToFront is " + minToFront(list));
}

public static ArrayList<Integer> minToFront(ArrayList<Integer> list) {
    int min = list.get(0);
    int i = 1;
    //find min value
    while (i < list.size()) {
        if (min >= list.get(i)) {
            min = list.get(i);
        }
        i++;
    }

    //insert to front
    int index = list.indexOf(min);
    list.remove(index);
    list.add(0, min);

    return list;
}
```

Console output:

The intial list is [2, 45, 1, 3]

The list after minToFront is [1, 2, 45, 3]

Problem 15

Problem statement: create a filterRange method

Code:

```
public static void main(String[] args) {
    ArrayList<Integer> alist = new ArrayList<Integer>();
    alist.add(1);
```

```

        alist.add(3);
        alist.add(5);
        alist.add(8);
        System.out.println("The intial unmodified list is " + alist);
        System.out.println("The list after fileterRange between 5 and 6 is " +
            filterRange(alist,5, 6));
    }

    //filterRange
    public static ArrayList<Integer> filterRange(ArrayList<Integer> list, int min, int
max) {
        ArrayList<Integer> newList = new ArrayList<Integer>();

        for(int item : list){
            if(item < min || item > max){
                newList.add(item);
            }
        }

        return newList;
    }
}

```

Output console:

The initial unmodified list is [1, 3, 5, 8]

The list after fileterRange between 5 and 6 is [1, 3, 8]

Problem 17

Problem Statement: Create an interleave method to put a2 arraylist into a1 arraylist

Code:

```

public static void main(String[] args) {

    //interleave
    ArrayList<Integer> a1 = new ArrayList<Integer>();
    ArrayList<Integer> a2 = new ArrayList<Integer>();
    a1.add(1);
    a1.add(2);

    a2.add(11);
    a2.add(22);
    a2.add(33);
    //when a2 is longer than a1
    System.out.println("New a1 is " + interleave(a1, a2));
}

```

```

        //when a2 has same length as a1
        a1.removeAll(a1);
        a1.add(1);
        a1.add(2);
        a1.add(3);
        System.out.println("New a1 is " + Interleave(a1, a2));

        //when a1 is longer than a2
        a1.removeAll(a1);
        a1.add(1);
        a1.add(2);
        a1.add(3);
        a1.add(4);
        a1.add(5);
        System.out.println("New a1 is " + Interleave(a1, a2));
    }

    //Interleave
    public static ArrayList<Integer> Interleave(ArrayList<Integer> a1, ArrayList<Integer>
a2) {

        System.out.println("The initial a1 is " + a1);
        System.out.println("a2 is " + a2);

        int a1Size = a1.size();
        int a2Size = a2.size();

        //a1 has same length a2
        if(a1.size() == a2.size()){
            a1.addAll(a2);
            for(int i = a2.size() - 1; i > 0; i--){
                a1.remove(i*2);
                a1.add(i*2, a1.get(i));
            }

            for(int i = a2.size() - 1; i >= 0; i--){
                a1.remove(i*2+1);
                a1.add(i*2+1, a2.get(i));
            }
            //a1 is shorter than a2
        } else if(a1.size() < a2.size()) {
            a1.addAll(a2);
            for (int i = a1Size - 1; i > 0; i--) {
                a1.remove(i * 2);
                a1.add(i * 2, a1.get(i));
            }
            for (int i = a2.size() - 1; i >= 0; i--) {
                if (i <= (a1Size - 1)) {
                    a1.remove(i * 2 + 1);
                    a1.add(i * 2 + 1, a2.get(i));
                } else {
                    a1.remove(i + a1Size);
                    a1.add(i + a1Size, a2.get(i));
                }
            }
        }
        } else{
            a1.addAll(a2);
            //Fill a1's values to new a1
            for (int i = a1Size - 1; i > 0; i--) {
                if (i <= (a2Size - 1)) {

```

```

        a1.remove(i * 2);
        a1.add(i * 2, a1.get(i));
    } else {
        a1.remove(i + a2Size);
        a1.add(i + a2Size, a1.get(i));
    }
}
//Fill a2 to a1
for(int i = a2.size() - 1; i >= 0; i--){
    a1.remove(i*2+1);
    a1.add(i*2+1, a2.get(i));
}
}

return a1;
}

```

Output Console:

The initial a1 is [1, 2]

a2 is [11, 22, 33]

New a1 is [1, 11, 2, 22, 33]

The initial a1 is [1, 2, 3]

a2 is [11, 22, 33]

New a1 is [1, 11, 2, 22, 3, 33]

The initial a1 is [1, 2, 3, 4, 5]

a2 is [11, 22, 33]

New a1 is [1, 11, 2, 22, 3, 33, 4, 5]