

## Lab 07 Assignment

班级：202111

学号：19241027

姓名：胡峻诚

### Question1 简述 ArrayList 和 LinkedList 的区别 简答

类似于 C 语言的数组和链表。

1. ArrayList 基于动态数组实现的非线程安全的集合；LinkedList 基于链表实现的非线程安全的集合。
2. 对于随机 index 访问的 get 和 set 方法，一般 ArrayList 的速度要优于 LinkedList。因为 ArrayList 直接通过数组下标直接找到元素；LinkedList 要移动指针遍历每个元素直到找到为止。
3. 新增和删除元素，一般 LinkedList 的速度要优于 ArrayList。因为 ArrayList 在新增和删除元素时，可能扩容和复制数组；LinkedList 实例化对象需要时间外，只需要修改指针即可。
4. LinkedList 集合不支持高效的随机随机访问（RandomAccess）
5. ArrayList 的空间浪费主要体现在在 list 列表的结尾预留一定的容量空间，而 LinkedList 的空间花费则体现在它的每一个元素都需要消耗相当的空间
6. ArrayList 是实现了基于动态数组的数据结构，LinkedList 基于链表的数据结构。
7. 对于随机访问 get 和 set，ArrayList 觉得优于 LinkedList，因为 LinkedList 要移动指针。
8. 对于新增和删除操作 add 和 remove，LinkedList 比较占优势，因为 ArrayList 要移动数据。

### Question2 写出以下程序的输出 简答

```
1 | I am an animal
2 | I am a dog
3 | I am an animal
4 | I am type T and my type is Animal
5 | I am type S and my type is Dog
```

The screenshot shows an IDE window titled 'Lab06 - AnimalTest.java'. The main editor displays the following Java code:

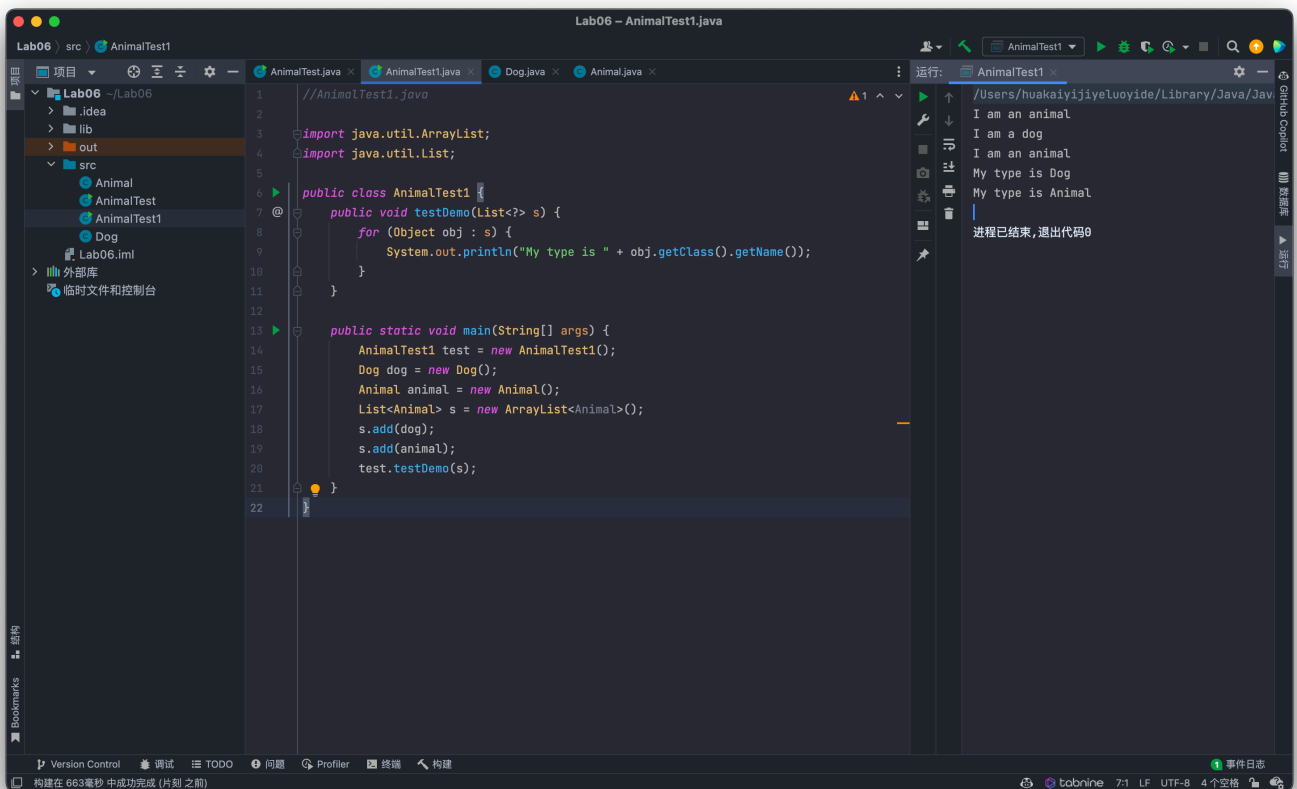
```
//AnimalTest.java
1 public class AnimalTest {
2     @
3     public <T, S extends T> T testDemo(T t, S s) {
4         System.out.println("I am type T and my type is " + t.getClass().getName());
5         System.out.println("I am type S and my type is " + s.getClass().getName());
6         return t;
7     }
8
9     public static void main(String[] args) {
10        AnimalTest test = new AnimalTest();
11        Dog dog = new Dog();
12        Animal animal = new Animal();
13        Animal animal1 = test.testDemo(animal, dog);
14    }
15 }
```

The right-hand pane shows the output of the program:

```
AnimalTest
I am an animal
I am a dog
I am an animal
I am type T and my type is Animal
I am type S and my type is Dog
进程已结束,退出代码0
```

### Question3 写出以下程序的输出(注意通配符的使用) 简答

- 1 I am an animal
- 2 I am a dog
- 3 I am an animal
- 4 My type is Dog
- 5 My type is Animal



## Question4 编写程序完成以下要求 编程

```

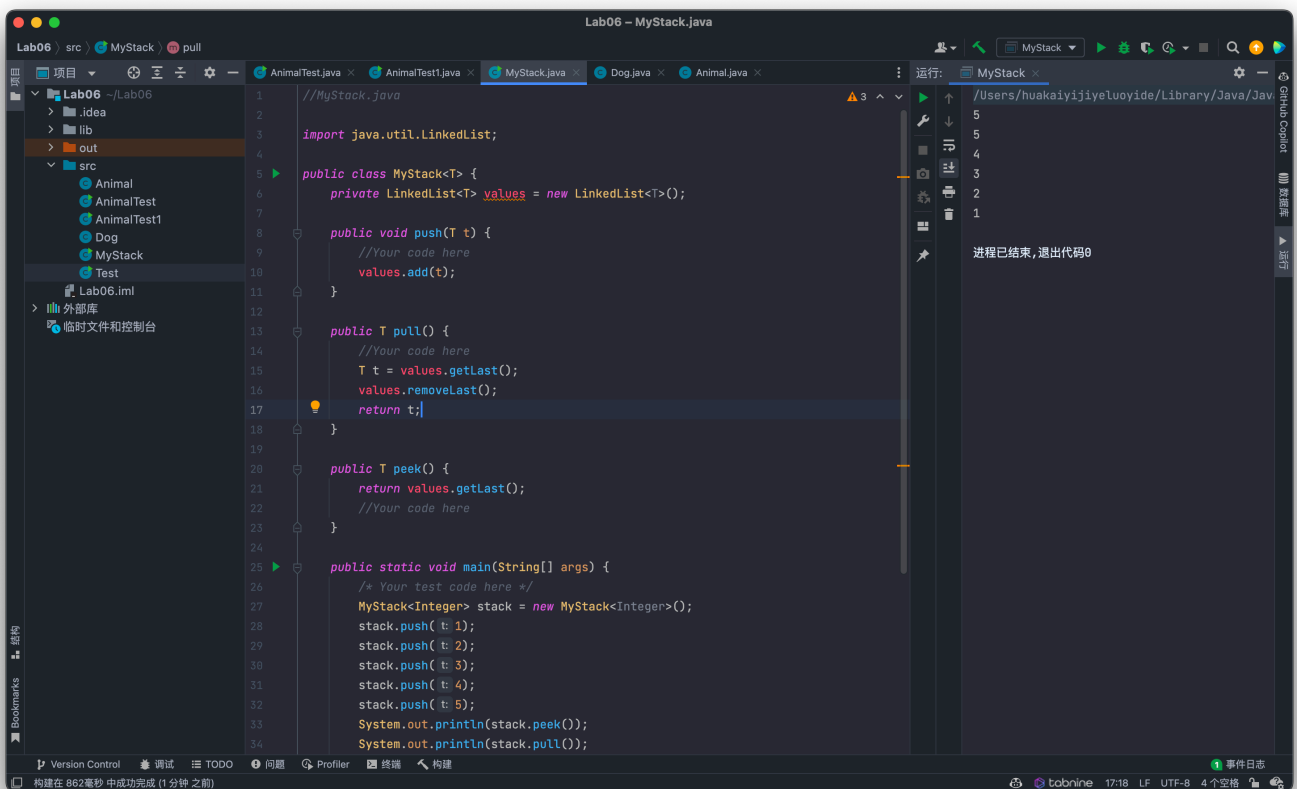
1 //MyStack.java
2
3 import java.util.LinkedList;
4
5 public class MyStack<T> {
6     private LinkedList<T> values = new LinkedList<T>();
7
8     public void push(T t) {
9         //Your code here
10        values.add(t);
11    }
12
13    public T pull() {
14        //Your code here
15        T t = values.getLast();
16        values.removeLast();
17        return t;
18    }
19
20    public T peek() {
21        //Your code here
22        return values.getLast();

```

```

23     }
24
25     public static void main(String[] args) {
26         /* Your test code here */
27         MyStack<Integer> stack = new MyStack<Integer>();
28         stack.push(1);
29         stack.push(2);
30         stack.push(3);
31         stack.push(4);
32         stack.push(5);
33         System.out.println(stack.peek());
34         System.out.println(stack.pull());
35         System.out.println(stack.pull());
36         System.out.println(stack.pull());
37         System.out.println(stack.pull());
38         System.out.println(stack.pull());
39     }
40 }

```



## Question5 编写程序完成以下要求

编程

```

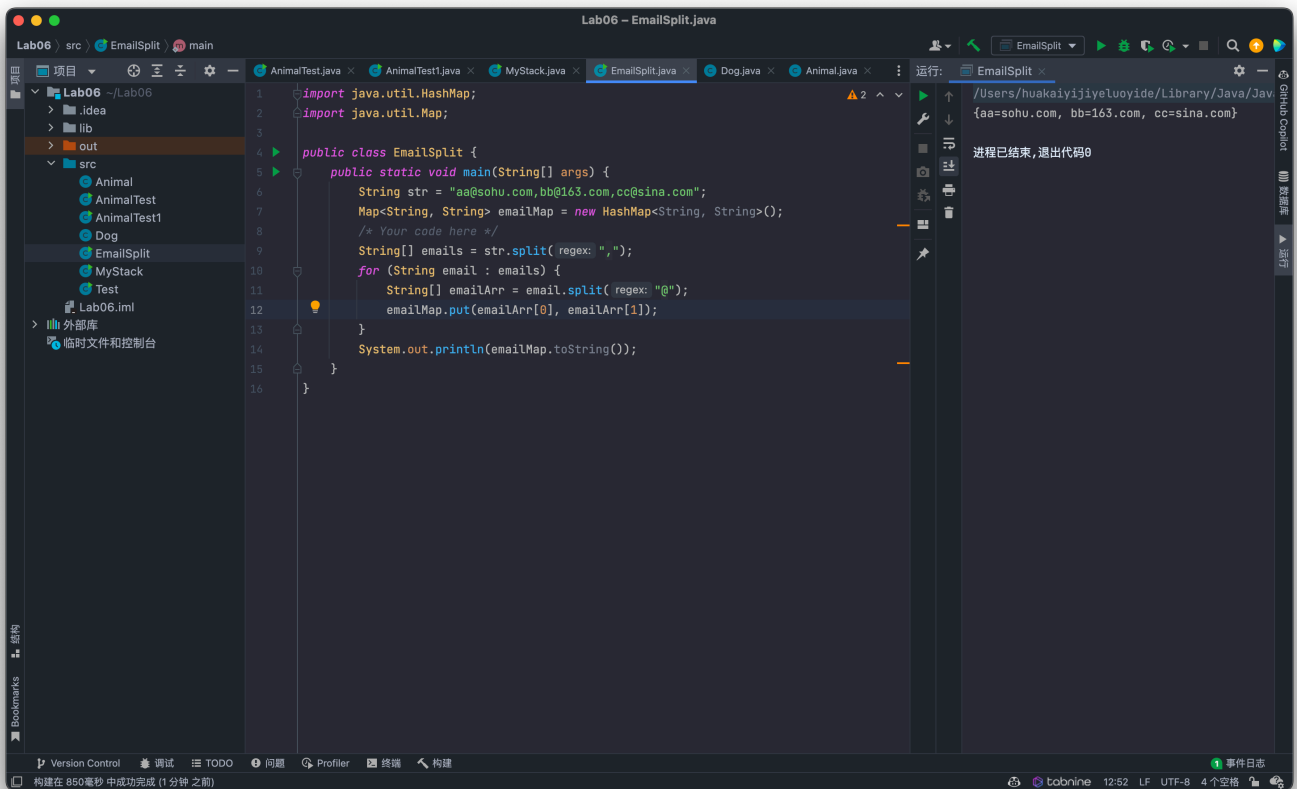
1 import java.util.HashMap;
2 import java.util.Map;

```

```

3
4 public class EmailSplit {
5     public static void main(String[] args) {
6         String str = "aa@sohu.com,bb@163.com,cc@sina.com";
7         Map<String, String> emailMap = new HashMap<String, String>();
8         /* Your code here */
9         String[] emails = str.split(",");
10        for (String email : emails) {
11            String[] emailArr = email.split("@");
12            emailMap.put(emailArr[0], emailArr[1]);
13        }
14        System.out.println(emailMap.toString());
15    }
16 }

```



## Question6 编写程序完成以下要求 编程

```

1 import java.util.HashMap;
2 import java.util.Map;
3
4 public class TwoColorBall {
5     public static void main(String[] args) {
6         /* Your code here */

```

```

7      Map<Integer, String> map = new HashMap<>();
8      while (map.size() < 6) {
9          int num = (int) (Math.random() * 33) + 1;
10         if (!map.containsKey(num)) {
11             map.put(num, "红球");
12         }
13     }
14     int num = (int) (Math.random() * 16) + 1;
15     map.put(num, "蓝球");
16
17     for (Map.Entry<Integer, String> entry : map.entrySet()) {
18         System.out.println(entry.getValue() + " " + entry.getKey());
19     }
20 }
21 }

```

