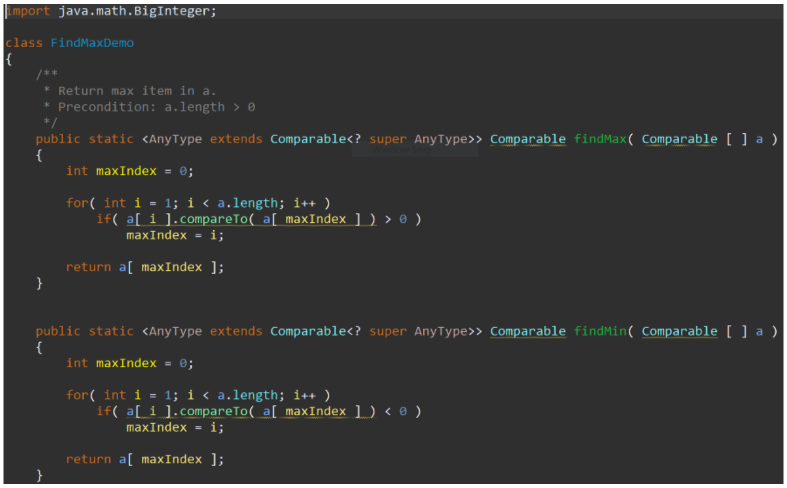
CS 2420
Fall semester section 01
Ricardo Gonzalez Mendez

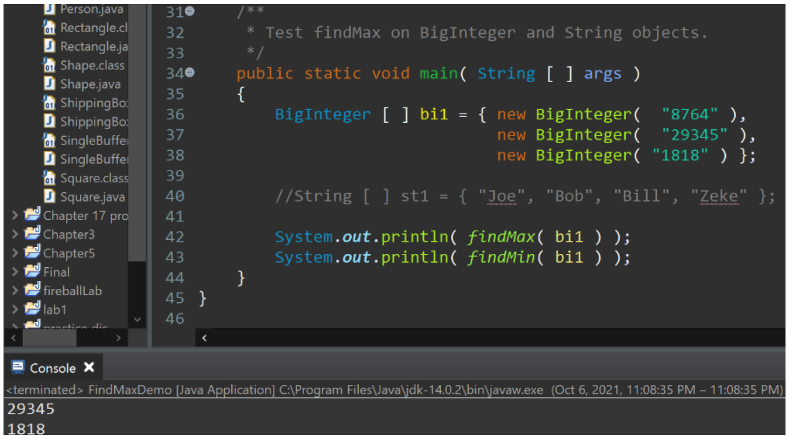
Chapter 4 Generics

4.23 and 4.21:

here is the code for the max and min and also the main with the test which shows that it is working. I found the max and min of the list and it was correct with the outputs I had.



```java
import java.math.BigInteger;

class FindMaxDemo
{
    /**
     * Return max item in a.
     * Precondition: a.length > 0
     */
    public static <AnyType extends Comparable<? super AnyType>> Comparable findMax( Comparable [ ] a )
    {
        int maxIndex = 0;

        for( int i = 1; i < a.length; i++ )
            if( a[ i ].compareTo( a[ maxIndex ] ) > 0 )
                maxIndex = i;

        return a[ maxIndex ];
    }


    public static <AnyType extends Comparable<? super AnyType>> Comparable findMin( Comparable [ ] a )
    {
        int maxIndex = 0;

        for( int i = 1; i < a.length; i++ )
            if( a[ i ].compareTo( a[ maxIndex ] ) < 0 )
                maxIndex = i;

        return a[ maxIndex ];
    }
```

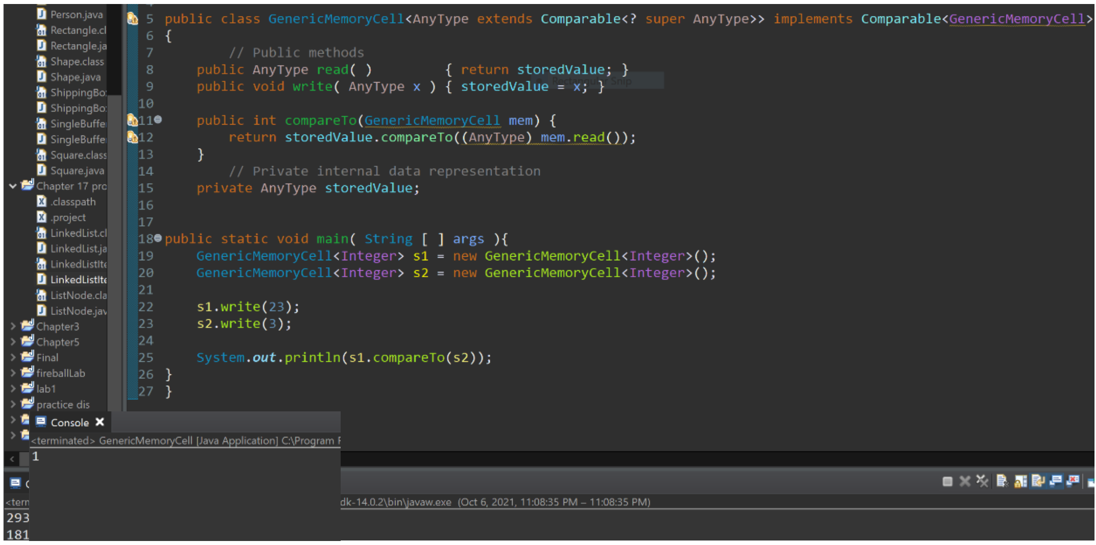


```java
    /**
     * Test findMax on BigInteger and String objects.
     */
    public static void main( String [ ] args )
    {
        BigInteger [ ] bi1 = { new BigInteger(  "8764" ),
                               new BigInteger(  "29345" ),
                               new BigInteger( "1818" ) };

        //String [ ] st1 = { "Joe", "Bob", "Bill", "Zeke" };

        System.out.println( findMax( bi1 ) );
        System.out.println( findMin( bi1 ) );
    }
}
```

```
<terminated> FindMaxDemo [Java Application] C:\Program Files\Java\jdk-14.0.2\bin\javaw.exe  (Oct 6, 2021, 11:08:35 PM – 11:08:35 PM)
29345
1818
```

4.29:

Here is the code and I test it with a main to see if the answer was right and it was since they weren't the same and s1 was not the same then it returns a 1.



```java
public class GenericMemoryCell<AnyType extends Comparable<? super AnyType>> implements Comparable<GenericMemoryCell>
{
        // Public methods
    public AnyType read( )         { return storedValue; }
    public void write( AnyType x ) { storedValue = x; }

    public int compareTo(GenericMemoryCell mem) {
        return storedValue.compareTo((AnyType) mem.read());
    }
        // Private internal data representation
    private AnyType storedValue;


public static void main( String [ ] args ){
    GenericMemoryCell<Integer> s1 = new GenericMemoryCell<Integer>();
    GenericMemoryCell<Integer> s2 = new GenericMemoryCell<Integer>();

    s1.write(23);
    s2.write(3);

    System.out.println(s1.compareTo(s2));
}
}
```

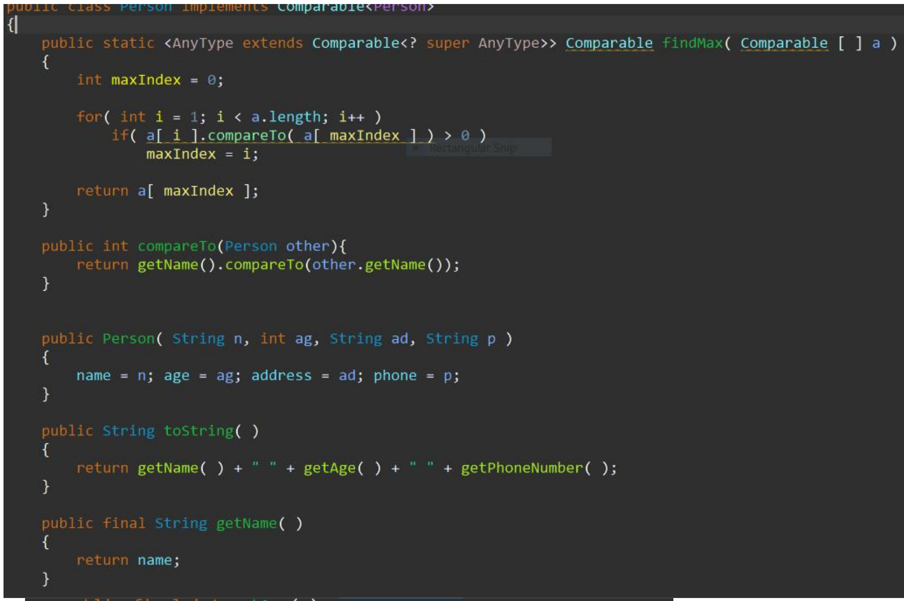
```
<terminated> GenericMemoryCell [Java Application] C:\Program F
1
```

```
<term                    dk-14.0.2\bin\javaw.exe  (Oct 6, 2021, 11:08:35 PM – 11:08:35 PM)
293
181
```

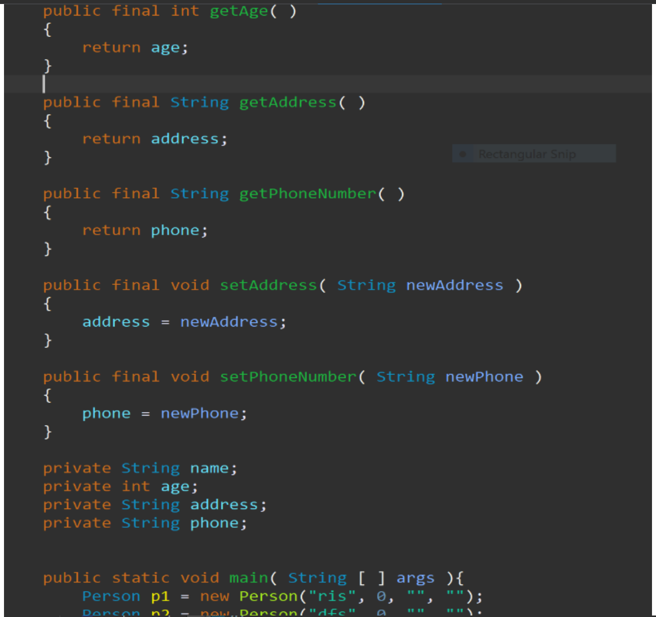
CS 2420
Fall semester section 01
Ricardo Gonzalez Mendez

4.36:

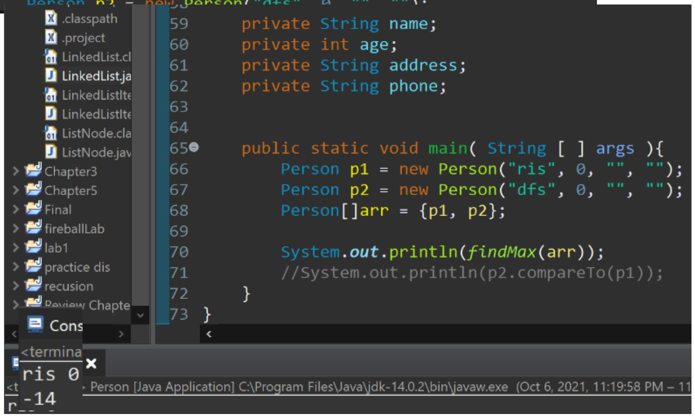
The code with person also the main to test out everything and it works. Also do test compare too. I am getting the right answer only thing I am returning -14 instead of just -1.



```java
public class Person implements Comparable<Person>
{
    public static <AnyType extends Comparable<? super AnyType>> Comparable findMax( Comparable [ ] a )
    {
        int maxIndex = 0;

        for( int i = 1; i < a.length; i++ )
            if( a[ i ].compareTo( a[ maxIndex ] ) > 0 )
                maxIndex = i;

        return a[ maxIndex ];
    }

    public int compareTo(Person other){
        return getName().compareTo(other.getName());
    }


    public Person( String n, int ag, String ad, String p )
    {
        name = n; age = ag; address = ad; phone = p;
    }

    public String toString( )
    {
        return getName( ) + " " + getAge( ) + " " + getPhoneNumber( );
    }

    public final String getName( )
    {
        return name;
    }
```



```java
    public final int getAge( )
    {
        return age;
    }

    public final String getAddress( )
    {
        return address;
    }

    public final String getPhoneNumber( )
    {
        return phone;
    }

    public final void setAddress( String newAddress )
    {
        address = newAddress;
    }

    public final void setPhoneNumber( String newPhone )
    {
        phone = newPhone;
    }

    private String name;
    private int age;
    private String address;
    private String phone;


    public static void main( String [ ] args ){
        Person p1 = new Person("ris", 0, "", "");
        Person p2 = new Person("dfs", 0, "", "");
```



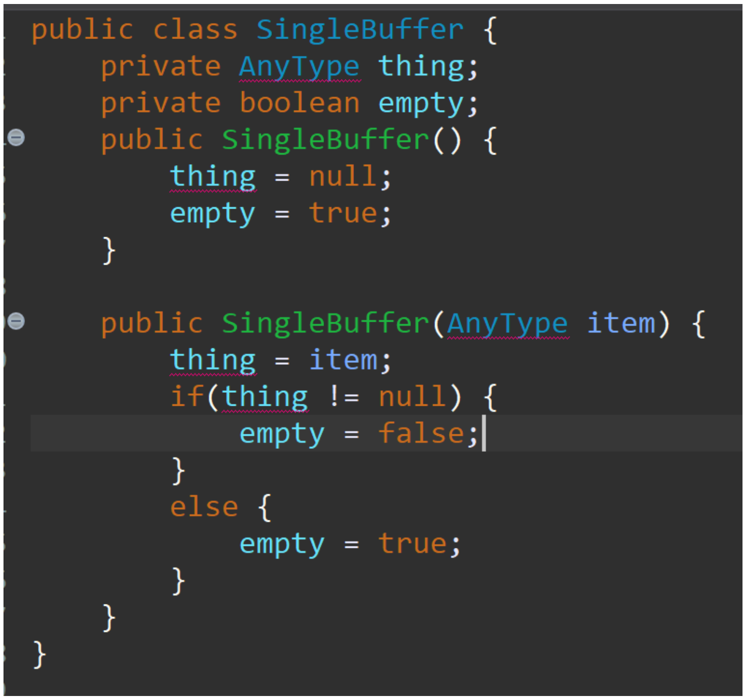
```java
    private String name;
    private int age;
    private String address;
    private String phone;


    public static void main( String [ ] args ){
        Person p1 = new Person("ris", 0, "", "");
        Person p2 = new Person("dfs", 0, "", "");
        Person[]arr = {p1, p2};

        System.out.println(findMax(arr));
        //System.out.println(p2.compareTo(p1));
    }
}
```

Person [Java Application] C:\Program Files\Java\jdk-14.0.2\bin\javaw.exe (Oct 6, 2021, 11:19:58 PM – 11

```
ris 0
-14
```

4.37:

Honestly didn’t work and I’m not to sure how to do it. But this is what I had



```
public class SingleBuffer {
    private AnyType thing;
    private boolean empty;
    public SingleBuffer() {
        thing = null;
        empty = true;
    }

    public SingleBuffer(AnyType item) {
        thing = item;
        if(thing != null) {
            empty = false;
        }
        else {
            empty = true;
        }
    }
}
```