1. (Bonus 2 points) What is your section?

   101

   101E

Page 1 _____ / 10

Page 2 _____ / 10

Page 3 _____ / 18

Page 4 _____ / 17

Page 5 _____ / 25

Page 6 _____ / 20

Total _____ /100

2. (5 points) What is the output of the following code segment?

```
int i = 0;
while ( i > 0 ) {
   ++i;
   System.out.println("true");
}
```

3. (5 points) What is the output of the following code segment?

```
for ( int i = 0; i < 2; ++i ) {
   for ( int j = 0; j < 2; ++j ) {
        break;
   }
   System.out.println ("outer");
}
System.out.println ("finished");
```

4. (5 points) What is the output of the following code segment?

```
int counter1 = 0;
int counter2 = 0;
for ( int i = 0; i < 5; ++i ) {
   for ( int j = 0; j < 10; ++j ) {
      ++counter1;
   }
   ++counter2;
}
System.out.println ("counter1 = " + counter1);
System.out.println ("counter2 = " + counter2);
```

5. (5 points) Write a code segment using a while loop that performs a comparable task as the following code segment. Assume `Scanner` variable `stdin` has already been defined.

```
int sum = 0;
for ( int i = 0; i < 10; ++i ) {
   int number = stdin.nextInt();
   sum = sum + number;
}
System.out.println(sum);
```

6. (5 points) What is the compiler error in the following code segment.

```
int[] array = { 8, 7, 6, 5, 4, 3 };
int sum = 0;
for ( int i = 0; i < array.length; ++i ) {
    sum = sum + array[i];
}
if ( i == sum ) {
   System.out.println("Same");
} else {
   System.out.println("Different");
}
```

7. (5 points) Suppose class `Widget` has an `int` instance variable `myData`. Suppose that `x` and `y` are separately initialized `Widget` variables. Is it necessarily the case that `x.myData` must always equal `y.myData`? Why? (Use no more than TWENTY words to explain why).

8. (5 points) Suppose class `Widget` has a `static int` member variable `theCount`. Suppose that `x` and `y` are separately initialized `Widget` variables. Is it necessarily the case that `x.theCount` must always equal `y.theCount`? Why? (Use no more than TWENTY words to explain why).

9. (3 points) Draw a memory diagram for the following definition.

```
int[] array;
```

10. (3 points) Draw a memory diagram for the following definition.

```
int[] array = new int[2];
```

11. (3 points) Draw a memory diagram for the following definition.

```
int[] array = { 1, 2 };
```

12. (3 points) Draw a memory diagram for the following definition.

```
String[] array = new String[2];
```

13. (3 points) Draw a memory diagram for the following definition.

```
String[] array = { "A", "B" };
```

14. (5 points) Why do our examples in lecture define `Scanner` variable `stdin` in the following way

```
Scanner stdin = new Scanner(System.in);
```

rather than this way

```
Scanner stdin = Scanner.create(System.in);
```

For questions 15–18 suppose class `Coordinate` has the following definition.

```
public class Coordinate {
   int x;
   int y;

   public Coordinate(int a, int b) {
      this.x = a;
      this.y = b;
   }
}
```

15. (5 points) Explain in TWENTY words or less why a `Coordinate` object has a `toString()` method.

16. (10 points) Define a public `clone()` member method with return type `Object` that returns a new `Coordinate` whose `x` and `y` values are the same as the `x` and `y` values of the `this` object.

17. (10 points) Define a `boolean` public method `equals()` that has a single `Object` parameter named `v`. The method should return false is `v` is not an *instance of* `Coordinate`. If `v` is a `Coordinate` then the method returns true if its `x` and y values match the `x` and `y` values of the `this` object; otherwise, the method returns false. Hint: if `v` is an instance of `Coordinate` then define a `Coordinate` variable `c` that equals a `Coordinate`-casted version of `v`.

18. (5 points) Write a single statement that when added to the `Coordinate` definition gives it a public class (static) constant named `ORIGIN` of type `Coordinate`. Constant `ORIGIN` should have both its `x` and `y` values being 0.

19. (10 points) Write a public static method `maximum()` with a return type of `int` that takes a single `int` array parameter named `data`. Method `maximum()` is to return the value of the largest element in `data`. You may assume that `data` has at least one element.

20. (5 points) Suppose class `ArrayManip` has an `int` array instance variable named `data`. Write a public member method `minimum()` with a return type of `int` for `ArrayManip`. Method `minimum()` is to return the value of the smallest element in the `data` array of the `this` object. You may assume that `data` has at least one element.

PLEDGE: