Pledged CS 101 Exam 3 Spring 2005 Email Id __________ Name ___________________________________

This pledged exam is open text book and closed notes. Different questions have different points associated with them. Because your goal is to maximize your number of points, we recommend that you do not dwell too long on any particular question during your first pass through the exam.

Unless specified all arrays are one-dimensional. Assume `stdin` is an initialized `Scanner` variable throughout the test.

Standard Java class `Point` for representing (x, y) locations is used throughout the test. Class `Point` has many methods. For our purposes assume the following public `Point` constructor and methods are always available.

- `Point()`: Configures a new `Point` object to represent the location (0, 0).
- `Point(int a, int b)`: Configures a new `Point` object to represent the location (`a`, `b`) .
- `int getX()`: returns the x-coordinate of its object.
- `int getY()`: returns the y-coordinate of its object.
- `void setX(int v)`: sets the x-coordinate of its object to value `v`.
- `void setY(int v)`: sets the y-coordinate of its object to value `v`.
- `Object clone()`: returns a new `Point` object that is a duplicate of its object.

Page 1	______ /	3
Page 2	______ /	23
Page 3	______ /	16
Page 4	______ /	14
Page 5	______ /	12
Page 6	______ /	10
Page 7	______ /	6
Page 8	______ /	16
Total	______ /	100

Pledge:

1. (3 points) What section of CS101 are you in?

_______ 2 `CS101E`

_______ 3 `0800-0915 Thursday`

_______ 4 `0930-1045 Thursday`

_______ 5 `1100-1215 Thursday`

_______ 6 `1230-1345 Thursday`

_______ 7 `1400-1515 Thursday`

_______ 8 `1530-1645 Thursday`

_______ 9 `1700-1815 Thursday`

_______ 10 `1830-1945 Thursday`

_______ 11 `2000-2115 Thursday`

2. (3 points) Assume `int` array variable `score` has already been defined and initialized. Write a *single* statement that assigns the value 1 to the first element of array `score`.

3. (3 points) Assume `int` array variable `score` has already been defined and initialized. Write a *single* statement that assigns the value 1 to the last element of array `score`.

4. (3 points) Assume `int` array variable `data` has already been defined and initialized. Fill in the blanks in the following code segment that sets the value of variable `total` to the sum of the values in `int` array `data`.

```
int total = 0;
for (int i = 0; ______________________ ; ++i) {

    ____________________________________________________________

}
```

5. (3 points) Write a single statement that defines and initializes an `int` array variable `b` to reference a new array with 10 elements.

6. (3 points) Assume `int` array variable `c` has already been defined and initialized. Write a single statement that defines and initializes an `int` array variable `b` to reference the same array of `int` elements as array `c`.

Suppose the following code segment occurs in the body of some method.

```
int[] a1;
int[] a2 = null;
int[] a3 = new int[0];
int[] a4 = new int[2];
```

7. (2 points) **TRUE** **FALSE** Variable `a1` references an array.

8. (2 points) **TRUE** **FALSE** Variable `a2` references an array.

9. (2 points) **TRUE** **FALSE** Variable `a3` references an array.

10. (2 points) **TRUE** **FALSE** Variable `a4` references an array.

11. (4 points) Fill in the blanks in the following code segment that displays the number of integer extracted (read) from standard input. Assume `Scanner` variable `stdin` was initialized using the definition on page 1.

```
int count = _____________________ ;

System.out.println("Enter a list of numbers using CTRL-Z to indicate");
System.out.println("when there are no more values");


while ( __________________________________________ ) {
   int number = stdin.nextInt();


   ___________________________________________________________;
}

System.out.println( count );
```

12. (4 points) Fill in the blanks in the following code segment that displays each of the integer values it was able to extract (read) from standard input. Assume `Scanner` variable `stdin` was initialized using the definition on page 1.

```
System.out.println("Enter a list of numbers using CTRL-Z to indicate");
System.out.println("when there are no more values");


while ( __________________________________________ ) {
   int number = stdin.nextInt();


   ___________________________________________________________;
}
```

13. (4 points) In twenty words or less explain why the following legal code segment may not produce any output. Assume `Scanner` variable `stdin` was initialized using the definition on page 1.

```
while ( stdin.hasNext() ) {
   String s = stdin.nextLine();
   System.out.println ( s );
}
```

14. (4 points) In twenty words or less explain why the while loop in the following legal code segment may not terminate. Assume `Scanner` variable `stdin` was initialized using the definition on page 1.

```
while ( stdin.hasNext() ) {
   System.out.println ( "Rinse, lather, repeat" );
}
```

15. (5 points) Fill in the blanks in the definition for boolean method hasIt(). The method takes a single int array parameter named data and returns whether or not an element of array data has the value 101. You may assume that data has at least one element.

```
public boolean hasIt( ____________________________________ ) {

   for (int i = 0; i < ____________________ ; ++i ) {

      if ( ____________________________________ ) {

        ____________________________________________________;
      }

   }

   ____________________________________________________________ ;
}
```

16. (4 points) Fill in the blanks in the definition for method makeArray(). The method at takes a single int parameter named n. The method returns a new int array with n elements.

```
public ____________________ makeArray( int n ) {

   ____________________________________________________________ ;

   return result;
}
```

17. (5 points) Fill in the blanks in the definition for method makeMatrix(). The method takes two int parameters named m and n respectively. Method makeMatrix() returns a new two-dimensional array with m rows where each row has n elements. In performing its task, the method makes use of method makeArray() from the previous question.

```
public ____________________ makeMatrix( int m, int n ) {

   ____________________________________________________________ ;

   for ( _____________________________________________________ ) {

      ____________________ = makeArray( ____________________ );

   }

   return result;
}
```

18. (4 points) Consider the following code segment.

```
int counter1 = 0;
int counter2 = 0;
int counter3 = 0;
int counter4 = 0;
for ( int i = 0; i < 10; ++i ) {
   ++counter1;
   for ( int j = 0; j < 10; ++j ) {
      ++counter2;
   }
   for ( int j = 0; j < 10; ++j ) {
      ++counter3;
   }
   ++counter4;
}
System.out.println ("counter1 = " + counter1);
System.out.println ("counter2 = " + counter2);
System.out.println ("counter3 = " + counter3);
System.out.println ("counter4 = " + counter4);
```

What output does it display?

counter1 = ______________________

counter2 = ______________________

counter3 = ______________________

counter4 = ______________________

In Questions 19–22 suppose only the following `display()` methods are available. Not all of the invocations in these questions are legal.

- `public void display(int x, double y)`: displays the *sum* of `x` and `y` to standard output.
- `public void display(int x, int y)`: displays the *product* of `x` and `y` to standard output.

19. (2 points) What is the result of attempting the invocation `display(4, 5)`.

20. (2 points) What is the result of attempting the invocation `display(4.0, 5.0)`.

21. (2 points) What is the result of attempting the invocation `display(4, 5.0)`.

22. (2 points) What is the result of attempting the invocation `display(4.0, 5)`.

23. (4 points) Complete the definition of `void` method `reverse()` that takes a single `int` array parameter named `x`. Method `reverse()` reverses the order of the element values in `x`. You may assume `x` has at least one element.

```
public void reverse ( ____________________________________________ ) {

   int n = _____________________________ ;
   for (int i = 0; i < n; ++i ) {

      ____________________________________________________________ ;
   }
}
```

24. (2 points) Consider the following code segment.

```
Point p = new Point(1, 1);
Point q = new Point(1, 1);
Point r = (Point) p.clone();
boolean flag1 = ( p == q );
boolean flag2 = ( p == r );
System.out.println( "flag 1 = " + flag1 );
System.out.println( “flag 2 = " + flag2 );
```

What output does it produce?

```
flag1 = ____________________

flag2 = ____________________
```

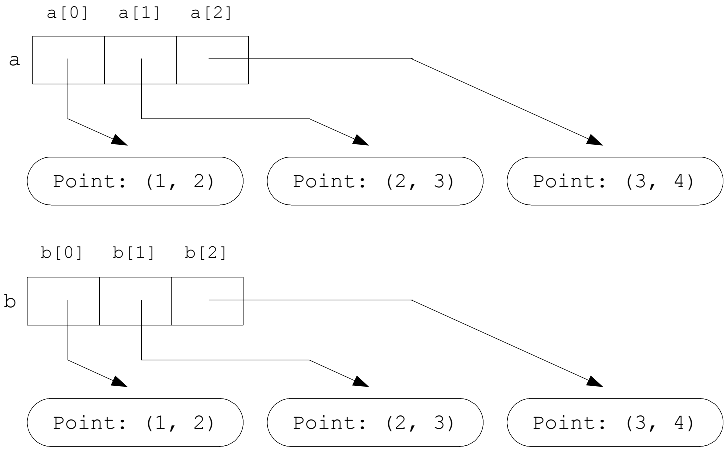
25. (4 points) Complete the definition of `void` method `fill()` that takes two parameters: an `int` array name `x` and `int` named `v`. Method `fill()` copies the value of `v` to all of the elements in `x`. You may assume `x` has at least one element.

```
public void fill ( ______________________________________________ ) {

   int n = _____________________________ ;
   for (int i = 0; i < n; ++i ) {


      ____________________________________________________________ ;
   }
}
```

26. (6 points) Complete the definition and initialization of a `Point` array named `b`. Array `b` is to be a deep copy of array `a`. A For example if array `a` represents the locations (1, 2), (2, 3), and (3, 4), then the memory diagram for arrays `a` and `b` would be



after the code completes. You may assume `a` has at least one element.

```
int n = a.length;

__________________________ b = ___________________________________ ;

for (int i = 0; i < n; ++i ) {

   int x = __________________________ ;


   int y = __________________________ ;

   Point ithDuplicate = new Point(x, y);

   __________________________ = ithDuplicate;
}
```

27. (4 points) Suppose `f()` is a `void` method that takes a single `int` as its parameter. In twenty words or less explain why 101 must be displayed by the following legal code segment.

```
int n = 101;
f(n);
System.out.println ( n );
```

28. (4 points) Suppose `g()` is a `void` method that takes a single `Point` as its parameter. In twenty words or less explain why 101 may not be displayed by the following legal code segment.

```
Point p = new Point(101, 101);
g(p);
System.out.println ( p.getX() );
```

29. (4 points) Suppose `h()` is a `void` method that takes a single `Point` as its parameter. In twenty words or less explain why *false* is always displayed by the following legal code segment.

```
Point p = new Point(101, 101);
Point q = (Point) p.clone();
h(p);
if ( p == q ) {
   System.out.println ( "true" );
}
else {
   System.out.println ( "false" );
}
```

30. (4 points) Suppose `i()` is a `void` method that takes a single `Point` as its parameter. In twenty words or less explain why 101 is always displayed by the following legal code segment.

```
Point p = new Point(101, 101);
Point q = (Point) p.clone();
i(p);
System.out.println ( q.getX() );
```