CIS 110 — Introduction to Computer Programming
Summer 2017 — Midterm

Name: _____

Recitation ROOM : _____

Pennkey (e.g., paulmcb): _____

My signature below certifies that I have complied with the University of Pennsylvania's Code of Academic Integrity in completing this examination.

_____ _____

Signature Date

Instructions:

- **Do not open this exam until told by the proctor**. You will have exactly 90 minutes to finish it.
- **Make sure your phone is turned OFF (not to vibrate!) before the exam starts.**
- Food, gum, and drink are strictly forbidden.
- **You may not use your phone or open your bag for <u>any</u> reason**, including to retrieve or put away pens or pencils, until you have left the exam room.
- This exam is *closed-book, closed-notes, and closed-computational devices*.
- If you get stuck on a problem, it may be to your benefit to move on to another question and come back later.
- All code must be written out in proper java format, including all curly braces and semicolons.
- **<u>Do not separate the pages.</u>** You may tear off the one scratch page at the end of the exam. This scratch paper must be turned in or you lose 5 points.
- Turn in all scratch paper to your exam. Do not take any sheets of paper with you.
- If you require extra paper, please use the backs of the exam pages or the extra pages provided at the end of the exam. **Only answers on the FRONT of pages will be grading. The back is for scratch work only.**
- Use a pencil, or blue or black pen to complete the exam.
- If you have any questions, raise your hand and a proctor will come to answer them.
- When you turn in your exam, you may be required to show ID. **If you forgot to bring your ID, talk to an exam proctor <u>immediately</u>.**
- We wish you the best of luck.

Scores: [For instructor use only]

Question 0		8 pts
Question 1		12 pts
Question 2		6 pts
Question 4		11 pts
Question 5		4 pts
Question 6		4 pts
Question 7		10 pts
Total:		**55 pts**

1.) Miscellaneous (8 points)

1.1) (1 point) The Easy One:

- Check that your exam has all 9 pages (excluding the cover sheet, which is no numbered).
- Write your name, recitation number, and PennKey (username) on the front of the exam.
- Sign the certification that you comply with the Penn Academic Integrity Code.

1.2) (3 points) In the space provided, draw the result of running the following code:

```
PennDraw.rectangle(0.75,0.5,0.25, 0.5);
PennDraw.circle(0.0,0.5,0.5);
```

1.3) (1 point) How many times would the following print HelloWorld?

```
int i = 15;
while (i > 1) {
    if (i % 2 == 0) {
       System.out.println("Hello World");
    }
    i /= 2;
}
```

a) **0 - Correct**

b) 1

c) 2

d) 3

e) Infinite (infinite loop)

f) 0, but with an infinite loop

1.4) (3 points) In the space below, write the above code using a for loop (the contents of the loop should not change, only it should be inside a for loop).

```
for (int i = 15; i > 1; i /= 2) {
    if (i % 2 == 0) {
        System.out.println("Hello World");
    }
}
```

2.) Operators and Expressions (12 points)

For each code fragment, (a) fill in the most appropriate word in the 1st column(if there are multiple blanks, all blanks are the same word) and (b) give the value that z contains after the code has been executed in the 2nd column.
If the code would result in an error, write "ERROR" in the 1st column and give the reason for the error in the 2nd column (you do not need to write the exact error message, just a general explanation). The first two problems have been completed for you.

`____??____ z = 11;` `z++;`	int	12
`int x = "Hello World";` `____??____ z = x.length();`	ERROR	Cannot set an int variable to a String

`String str = "25";`
 	
`int z = ____??____.parseInt(str);`
	Integer	25

`String s = "Catdog";`
 	
`____??____ z = s.charAt(6);`
	ERROR	OUT OF BOUNDS on the String

`____??____ z = {3, 2, 3, 4};`
 	
`int x = z[2];`
 	
`x++;`
	int[]	{3,2,3,4}

`____??____ z = 'A';`
 	
`z += 3;`
	char	'D'

`____??____ z = 3.0 + 7 / 2;`
`if (z = 6) {`
`  z++`
`}`
	ERROR	Should be == on line 2

`____??____[] x = new ____??____[4];`
 	
`x[3] = true;`
 	
`____??____ z = x[0];`
	boolean	false

3) (6 points) Tracing

The following code represents Collatz Conject. The (unproven) conjecture is that the loop will eventually terminate for every positive integer. For the given function calls, write what will be printed to the console.

```
1   public static void collatz(int num) {
2       while (num != 1) {
3           System.out.print(num + “,”);
4           if (num % 2 == 0) {
5               num /= 2;
6           } else {
7               num *= 3;
8               num++;
9           }
10      }
11      System.out.println();
12  }
```

ONLY WRITE WHAT PRINTS IN THE BOX

A) `collatz(8);`

```
8,4,2
```

B) `collatz(3);`

```
3,10,5,16,8,4,2
```

4) Nested Loops (11 points total)

The following code is used to print every **prime** number from 2 to 100 in ascending order. Fill in the blanks to complete this code.

```
for (int i ____________________A____________________) {
     boolean prime =___B___;
     for (int j ____________________C____________________) {
         if (i % j __D__ 0) {
             prime = __E__;
             ___F___; //efficiently leave inner loop
         }
     }
     if (__G__) {
          System.out.println(i) //i is prime
     }
}
System.out.println(); //print new line afterwards
```

A) = 2; i <= 100; i++
B) true
C) = 2; j <= Math.sqrt(i); j++
D) ==
E) false
F) break
G) prime

A and C are worth 3 points
B, D, E, F, G are worth 1 point

5) Method Writing – Easy (4 points)

You will be writing a **void** method that takes in an integer array (int[] arr) and two numbers (int a and int b). These numbers are indices of elements in the array. You will swap the values arr[a] and arr[b] within the array. This function should modify the array passed in as an argument.

You can ASSUME the following:

- The array has at least 2 elements
- a and b are valid indices (not out of bounds)
- a is not equal to b

Do not write any code to address those assumptions, just assume those things are all true.

```
public static void swap(int[] arr, int a, int b) {

    int temp = arr[a];
    arr[a] = arr[b];
    arr[b] = temp;
}
```

6) Method Testing (4 points)

Now that you have written your function, write a test for the function in main. This test should have specified all three elements we talked about in class. You may assume there is a `printArray(int[] arr)` method that prints the contents of the array in order. Example: `printArray(` [1,2,3]) would print as "1,2,3,"

```
public static void main(String[] args) {
   int[] arr = {1,2,3};
   swap(arr,1,2);
   System.out.println("The next line should print [1,3,2]");
   printArray(arr);
}
```

7) Method Writing – Harder (10 points)

Write a method that returns an integer that takes in an array of POSITIVE doubles and return the largest value possible by raising one value in the array to a different value in the array. For example, in [1,3,4], the largest possible exponent combination is 3^4 (which equals 81). This is larger than 4^3 (which equals 64), 4^1 (which equals 4), etc. **Note that 4^4 would not be a valid output from this function, since they reference the same value in the array.** You can use the `Math.pow(double a, double b),` which is a built in function that returns a^b (this works for non-integer bases and exponents).

```
public static double maxPower(double[] arr) {
     int maxPower = 0.0;
     for (int i = 0; i < arr.length; i++) {
          for (int j = 0; j < arr.length; j++) {
               if (i != j && maxPower < Math.pow(i, j)) {
                    maxPower = Math.pow(i, j);
               }
          }
     }
     return maxPower;
}
```

Extra Space for Answers

DO NOT RIP THIS PAGE OFF. ANY WRITING ON THIS PAGE CAN BE GRADED