CIS 110 — Introduction to Computer Programming
Fall 2018 — 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 3 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 1		6 pts
Question 2		14 pts
Question 3a		5 pts
Question 3b		8 pts
Question 4		15 pts
Question 5a		8 pts
Question 5b		4 pts
Question 6		
Total:		**62 pts**

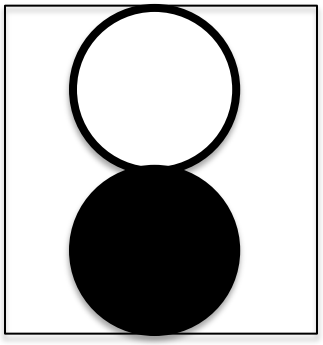
1.) Miscellaneous (6 points)

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) (2 points) In the space provided, write code to draw the figure pictured in the window to the right (you can assume the square is half as wide as the window it is in. You do not have to draw the window, only the shapes. You do not need to write the class or main declaration.

```
PennDraw.circle(0.5, 0.75, 0.25);
PennDraw.filledCircle(0.5, 0.25, 0.25);
```



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

```
int i = 22;
while (i > 0) {
    i /= 3;
    if (i % 2 == 0) {
        System.out.println("Hello World");
    }
}
```

2 Times

(2 points) Rewrite the above code using a for loop so it prints the same number of times. You are welcome to change the starting value of i, the exit condition, and/or the modification step. The goal is to print Hello World the same number of times.

```
for (int i = 0; i < 2; i++) {
    System.out.println("Hello World");
}
```

2.) Operators and Expressions (14 points) – 1 point for each blank

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.

DO NOT WRITE WHAT PRINTS. ONLY THE TYPE IN THE BLANK AND THE VALUE OF Z

`____ z = 1;` `z = z + 2;`	int	3
`int x = "Hello World";` `____ z = x.charAt(0);`	ERROR	Cannot set an int variable to a String

`String str = "hello world";`
 	
`____ z = s.length + "14";`
	ERROR	Strings have to use .length()

`int[] x = {1, 2, 3, 4, 5};`
 	
`____ z = x.length + "14";`
	String	"514"

`int[] x = {1, 2, 3, 4, 5};`
`String s = "taco salad";`
`____ z = x[2] + s.charAt(2) -`
`s.charAt(6);`
`//line 4 is a continuation of line 3`
	int	5

`int y = 5;`
 	
`____ z = y + "" + 4.0;`
	String	54.0

`int[] x = {2, 9, 1, 4, 6};`
`int[] snd = fst;`
`____ z = snd[1] % fst[4];`
	int	3

`____ x = "CIS110".charAt(0) == 'C';`
`____ y = 3 != 5;`
`____ z = (!x || y) && (y || (!y && x));`
	boolean	true

`____ z = 3 * (3 / 2) + 1.5;`	double	4.5

3a) Conditionals Fill-In (5 points total)

Fill in the blanks for the following method. This method takes in 3 numbers and returns true if they are all different. If any two match, or all 3 match, then it returns false.

```
public static boolean allDiff(double a, double b, double c) {


    if ( a != b ) { //2 points


        if ( b != c ) { //2 points

              return a != c;
        }
    }

    return false; //1 point
}
```

3b) (8 points) Loops Fill-In

Fill in the blanks for the following method. This function counts the number of odd numbers in an array.

```
public static int oddsInArray(int[] arr) {

     int count = 0; // 2 points


     for (int j = 0; j < arr.length; j++){ // 3 points


          if (arr[j] % 2 == 1) { // 2 points
               count++;
          }
     }

     return count; // 1 point

}
```

4) Tracing (15 points –0.5 points for each blank)
In the following code, print the value of x and y at each checkpoint. It is highly recommended you tear off and use the scratch page at the end of the exam

```
int[] x = {1, 5, 2, 11, 3};
int[] y = {2, 3, 3, 9, 4};

for (int i = 1; i < 5; i += i) {
    int k = 4 - i;
    while (x[i] > y[k]) {
         x[i] -= y[k];
    }
}
//Checkpoint A

for (int i = 0; i < 5; i++) {
     int j = x[i]*2;
     int k = j % 5;
     if (y[k] > x[i]) {
         y[k] += x[i];
         x[i] = y[k];
     }
}
//Checkpoint B

for (int i = 0; i < 5; i+=2) {
     if (x[i] >= y[i]) {
         y[i] = y[i] - x[i];
         x[i] = y[i] - x[i];
     } else {
         i--;
     }
}
//checkpoint C
```

	x	y
Checkpoint A	[1, 5, 2, 11, 1]	[2, 3, 3, 9, 4]
Checkpoint B	[4, 5, 6, 11, 5]	[2, 3, 5, 9, 6]
Checkpoint C	[-6, 5, -7, 11, 5]	[-2, 3, -1, 9, 6]

5a) Method Mixup (8 points)

Will is notoriously disorganized. It's gotten so bad he's actually gotten all of this lines of code out of order. The lines of code below belong to a method that search through an array create a new array with ONLY the even numbers. However, the lines are out of order. In the space at the bottom, put the lines of code below in the correct order. **You may not write your own code. You must use the code as written.**

```
}
}
}
}
}
count++;
for (int i = 0; i < arr.length; i++) {
for (int i = 0; i < arr.length; i++) {
if (arr[i] % 2 == 0) {
if (arr[i] % 2 == 0) {
insertAt++;
int count = 0;
int insertAt = 0;
int[] out = new int[count];
out[insertAt] = arr[i];
return out;
```

```
public class Exam {
        public static int[] getEvenArray(int[] arr) {
           //insert lines of code above to complete the method
        int count = 0;
        for (int i = 0; i < arr.length; i++) {
             if (arr[i] % 2 ==0) {
                  count++;
             }
        }

        int insertAt = 0;
        int[] out = new int[count];
        for (int i = 0; i < arr.length; i++) {
            if (arr[i] % 2 == 0) {
                 out[InsertAt] = arr[i];
                 insertAt++;
            }
        }

        return out;

      }
   }
```

5b) Method Mix-up Testing (4 points)

The method on the previous page takes in an array and returns an array that contains only the even numbers in the same order. Given this description, complete the Junit test below. You can assume that the method `getEvenArray` is in a file called Exam.java in the same folder as this program. (Hint: The method `assertArrayEquals(int[] a, int[]b)` returns true if the **contents** of the arrays `a` and `b` are equal. `assertEquals()` without the word Array does NOT work on arrays.) Write the test as we have shown in class. A test input has been generated for you.

```
import static org.junit.Assert.*;
import org.junit.*;

public class ExamTest {
    @Test
    public void getEvenArrayTest() {

        int[] inputArray = {5, 6, 2, 1, 4, 3};
        int[] expectedOutput = {6,2,4};
        int[] actualOutput = Exam.getEvenArray(inputArray);
        assertArrayEquals(expectedOutput, actualOutput);
    }
}
```

7) Method Writing – Harder (10 points)

MC BananaPeelZ, a soundcloud star, has recently been contacted by DMX to produce a cool hip and trendy rap song. While MC BananaPeelZ was very excited to work with a legend, he discovered something tragic: DMX does not write his own ad-libs. In fact, MC Banana Peelz found out that the positioning of the various "Come on!"s, "Yeah"'s and "Uh"'s throughout DMX songs are actually computer generated!

MC BananaPeelZ discovered that the algorithm works as follows

- 1 *YEAH* on every beat that is a power of 2 (the beat = 2^n for some n >= 0).
- 1 *uh-huh* per every 2 beats
- Finally, the song ends with a "Grrr… WOOF WOOF."

Here's an example of a series of adlibs created by DMX's adlib method, for a song with only 9 beats. We have annotated the beats with comments. The comments are NOT part of the program output. The method does not return anything, but simply prints.

uh-huh //beat 0

YEAH! //beat 1

YEAH! //beat 2

uh-huh //beat 2

YEAH! //beat 4

uh-huh //beat 4

uh-huh //beat 6

YEAH! //beat 8

uh-huh //beat 8

Grrr... WOOF WOOF //end of the song

Wanting to expose DMX as a computer generated hack, MC BananaPeelZ has tasked you with writing the code that performs this behavior on the next page. Write a function that produces DMX's "ad-libbing" on the next page.

```
public static void dmxAdLibPrinter(int totalBeats) {
        int nextPowerOfTwo = 1; //2 ^ 0

        for(int beat = 0; beat < totalBeats; beat++){
            //if beat is a power of two, print yeah
            if(beat == nextPowerOfTwo){
                nextPowerOfTwo *= 2;
                System.out.println("YEAH!");
            }
            //if even print uh-huh
            if(beat%2 == 0){
                System.out.println("uh-huh");
            }

        }
        //end the song
        System.out.println("Grrr... WOOF WOOF");
    }
```

Extra Space for Answers

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

Scratch Paper

YOU MAY USE THIS PAGE FOR SCRATCH WORK, BUT NOTHING ON IT WILL BE GRADED