

This exam is: **closed-book**, **NO electronic devices allowed**, and **closed-notes**. The exception is the “sage page” of the designated size on which you may have notes to consult during the exam.

Be sure you: **Provide legible answers in designated areas (credit will not be given for work that is difficult to read or not where expected)**, **Ensure you clearly fill in a single circle/square(s) on multiple choice questions**, **Use indentation of your code to show its structure (but don't dwell on exact punctuation/syntax)**, **Leave the exam stapled together in its original order**, **Do NOT attach any other pages to the exam**. You are welcome to use the blank space on the exam for any scratch work.

If you need to leave the room for any reason prior to turning in your exam, you must leave your exam and any electronic devices with a proctor. **We do not clarify or explain anything during the exam session. State your assumptions if something is unclear and do the best you can.**

Question:	1	2	3	4	5	6	7	8	Total
Points:	14	8	4	8	18	24	12	12	100

You must complete all the identifying information below correctly. Failure to do so is grounds for a zero on this exam:

1. Name (**print clearly**): _____
2. Student ID (**print clearly; 1 digit per underline**): _____
3. You must sign the pledge below for your exam to count. The penalty for cheating will be decided during academic integrity review, but the instructors will recommend an F in this course as the minimum penalty.

I have read the instructions on this page and I will neither give nor receive any unauthorized aid on this exam.

(Sign above)

⇒ Do not proceed until told to do so! ⇐

⇒ Initial the top right corner of each page before starting ⇐

1. (14 points) For each expression below indicate the type and value of the result. Use quotation marks to indicate strings. The first row has been completed as an example.

Expression	Result Type				Result Value
"Good "+"Luck!"	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input checked="" type="radio"/> String	"Good Luck!"
1+3+"1"	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
(!true) !(false)	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
1+9*2+"7"	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
((int)2.5)/2	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
(7/2.0)<3.5	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
15%4>15%3	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	
"so" + (4<3)	<input type="radio"/> boolean	<input type="radio"/> double	<input type="radio"/> int	<input type="radio"/> String	

2. (8 points) Indicate the most appropriate data type for each concept:

- (1) The number of alien spaceships you have seen:
 - boolean
 - double
 - int
 - String
- (2) The U.S. state where area 51 is located:
 - boolean
 - double
 - int
 - String
- (3) The speed of an alien spaceship:
 - boolean
 - double
 - int
 - String
- (4) Do you believe in aliens?:
 - boolean
 - double
 - int
 - String

3. (4 points) Given the following Java statement:

```
a = 3 > 9 + 7.0 * 2 && true;
```

(1) Which operation is completed first?

> + * &&

(2) Which operation is completed second?

> + * &&

(3) Which operation is completed third?

> + * &&

(4) Which of these operations is completed last?

> + * &&

4. (8 points) Fill in the boxes(s) whose loop code will properly print out all of the values of the following array:

```
int [] data = new int [10];
```

The values can be printed out in any order, as long as they are all printed. Fill in all that apply.

this code prints all of the values:

```
int i=10;
while (i>0) {
    i--;
    System.out.println(data[i]);
}
```

this code prints all of the values:

```
for (int i=0; i<=data.length-1; i++) {
    System.out.println(data[i]);
}
```

this code prints all of the values:

```
for (int i=9; i>0; i--) {
    System.out.println(data[i]);
}
```

this code prints all of the values:

```
int i=0;
while (i<data.length) {
    System.out.println(data[i]);
    i++;
}
```

5. Multiple choice. Fill in the correct circle or box(es):

(1) (4 points) Consider the following loop:

```
while(x <= 0 || x>5) {  
    // some code  
}
```

Which of the following conditions, if true, would cause the loop to continue running? Fill in all that apply.

$x = 3$ $x = -3$ $x \geq 6$ $x < -2$

(2) (2 points) Using the notation like (*typename*) in an expression, like the “(int)” in “(int) (Math.random()*2)” is called:

Casting Cromulescence Intification Retyping

(3) (2 points) Given the following Java statement:

```
int [][] array = new int [2][5];
```

If an array were not used, how many int variables would be needed to contain the same number of values as array can contain?:

1 2 3 5
 7 9 10 50

(4) (10 points) Given the following Java statement:

```
int [] array = {1, 2, 3, 4, 5};
```

For the following three parts assume a valid statement is one that will compile and execute without causing the program to “crash”.

i. Is `System.out.println(array[2/3])` a valid statement?

No Yes

ii. Is `System.out.println(array[10/2])` a valid statement?

No Yes

iii. Is `System.out.println(array[3.0/2.0])` a valid statement?

No Yes

iv. What will be printed by `System.out.println(array[1])`?

1 2 3 4

v. What are valid indices for the array (ranges are inclusive)?

[0-4] [0-5] [0-6] [1-4] [1-5] [1-6]

6. Show the output of each snippet of code in the area provided. Be as accurate as possible, including spacing and line usage. All output should start on the first dotted line on the right side of the paper (the leftmost dot corresponds to the left side of the console window). *The first line of the first problem has been filled in as an example. Correct answers may not use all lines. That is, there may be more space than needed.*

(1) (3 points)

```
System.out.println(0);
for(int i=1; i<3; ++i) {
    System.out.println(i);
}
```

0

.....

(2) (4 points)

```
int i = 1;
while(i<5) {
    i = 2*i;
    System.out.println(i);
}
```

.....

(3) (8 points)

```
for (int j=0; j<3; ++j) {
    for (int i=0; i<=j; ++i) {
        System.out.println(i + "." + j);
    }
}
```

.....

Show the output of each snippet of code in the area provided (continued)

(4) (5 points)

```
int i=5;
```

```
int j=1;
```

```
if (i > 5) {
```

```
    System.out.println(0);
```

```
    if (j < 5) {
```

```
        System.out.println(1);
```

```
    } else {
```

```
        System.out.println(2);
```

```
    }
```

```
    j = j + 7;
```

```
} else {
```

```
    System.out.println(3);
```

```
    if (j < 5) {
```

```
        System.out.println(4);
```

```
    } else {
```

```
        System.out.println(5);
```

```
    }
```

```
    j = j + 7;
```

```
}
```

```
System.out.println(6);
```

(5) (4 points)

```
double d=5;
```

```
while (d > 1.0) {
```

```
    System.out.println(d+0.1);
```

```
    d=d/2.0;
```

```
}
```

7. (12 points) Write code that will print information based on the integers from 1 to N inclusive. Assume N is a variable that is already set to some positive value. It should follow these rules:
- If the integer value is divisible by 3 print the word “pop”.
 - If the integer value is greater than 90 print the word “corn”.
 - If the integer value is both divisible by 3 and greater than 90, print only “popcorn”.
 - If none of the above are true just print the integer.

An example of the output follows:

```
1
2
pop
4
5
pop
.
.
.
88
89
pop
corn
corn
popcorn
corn
...
```

Show work on the next page.

Show work on the next page.

Show work on the next page.

Show work on the next page.

Show work on the next page.

