





Unit 2 – ICS3U0 – Java Loops and Methods

Sample Test – Thursday March 21, 2024

Name: Gorski

Total	%	Knowledge 	Communication 	Thinking 	Application 
(88)	(100)	(22)	(22)	(22)	(22)

Knowledge

1. (a) What are the parts of a loop? (Fill in the rest of the words)

/12

1	Initialize	Loop	stopping	variable
2	Test	Loop	stopping	condition
3	steps	To	Repeat	
4	Progress	To	The Loop	stopping condition

(b) On **both** of the following loops, circle each part of the loop. Use the numbers from (a) to label each part.

```

1 char again = 'y';
2 while (again != 'n') {
3     System.out.println("What a lovely day it is today.");
4     System.out.println("The sun is shining, and the birds are singing.");
5     System.out.println("How nice to have a sample test to cap it all off.");
6     again = IO.inputChar("Agreed? (y/n) ");
7 }
    
```

```

1 for (int i=0; i<100000; i++)
2     system.out.print("what?!? You don't enjoy writing sample tests? Huh.");
3     System.out.println();
    
```

← Not part of the loop. No Brackets.

2. Circle true or false based on the adjacent code.

/10

```

for (int i = 0 ; i < 5 ; i++){
    System.out.print (i * i + " ");
}
0 1 4 9 16
System.out.println ("");
    
```

```

for (int m = 128 ; m > 6 ; m=m/2){
    System.out.print (m + " ");
}
128 64 32 16 8
System.out.println ("");
    
```

- F a. The loop stopping condition in the first loop is $(i > 5)$. $i < 5$
- F b. The Boolean expression in the second loop is $(m = m/2)$.
- F c. There only are 2 loops. *Progress*
- F d. If the stopping condition in the first loop were changed to $i >= 0$, the first loop would not stop. *it is always true to the loop continues*
- F e. Without the code $i++$, the first loop would not stop. *i stays 0 forever*
- F f. This program contains repetition and decision control structures. *✓ Loop ✗ if*
- F g. All loop stopping variables need to be int types. *see char #1 b*
- F h. $i++$ adds 1 to the variable i and saves the result in i .
- F i. The first loop prints: 1 4 9 16 *missing 0*
- F j. The second loop prints: 128 64 32 16 8

Communication

3. Name four of Steven Johnson's factors that lead to creativity. ^{Not Eureka Moment. he argues that's wrong.} /2

Slow Hunch	Error	Platforms	Exaptation
------------	-------	-----------	------------

4. These questions relate to the PDLC. /8

(a) What does PDLC stand for?

Product
Development
Life
cycle

(b) What are the phases of the PDLC (in order)?

1	Analysis
2	Design
3	Code
4	Reflection

(c) Name the associated phase of the PDLC.

1	Create advertisements.	Reflection
2	Brainstorm a new game idea.	Analysis
3	Beta Testing.	Code
4	Write if statements.	Code
5	Write comments.	Code
6	Draw a structure chart.	Design

5. Fill in the terms described in the first column. /8

Method Signature	a) The most important line of a method; it contains the parameters, return type and method name.
Parameters	b) Values that are passed into a method.
Comments	c) Lines in your program that don't run.
Flow charts	d) Charts drawn to layout input, output, Boolean expressions and processing in the program.
Structure charts	e) Charts drawn to show the inputs and outputs to methods.
Testing	f) $\frac{3}{4}$ of the code phase is devoted to this.
Programmer	g) A job in the Coding Phase.
Script Writer	h) A job in the Design Phase.

6. Explain why Candy Crush's PDLC is significant in the history of games. (2 points, one sentence each) /2

Candy Crush's Design Phase had many clever choices. First they designed the colours, screens and characters appeal to diverse users; this increased their market. Also, they used the freemium model to increase sales. These choices made Candy Crush the first app to make \$1 m per day.

7. Why are methods useful? (2 points, one sentence each) /2

- ① Methods allow us to organize our code. For example, in our scavenger hunt program we put each place in a method.
- ② Methods allow us to repeat code by calling the method instead of cutting and pasting. We used this feature in our drawing methods codes to simplify the code.

Thinking

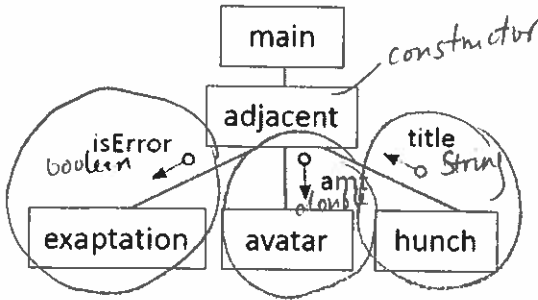
Return type
method name
parameter

8. True or False, circle the most correct answer concerning the method shown. /5

```
public boolean isLeapYear (int yr) {
    if (yr % 400 == 0)
        return true;
    else if (yr % 100 == 0)
        return false;
    else if (yr % 4 == 0)
        return true;
    else
        return false;
}
```

- T a) The method name is boolean. *isLeapYear*
- T b) The method return type is int. *boolean*
- T c) The parameter type is yr. *int*
- T d) The parameter name is isLeapYear. *yr*
- T e) This method would send out an char. *boolean*

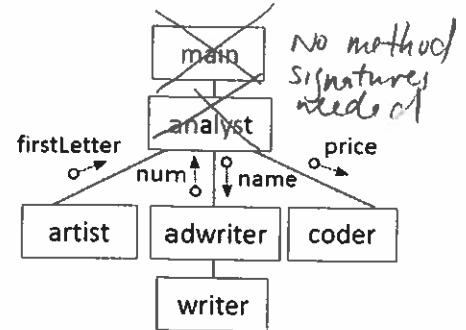
9. Answer the following short answer questions about this structure chart. /5



(a) How many methods?	3
(b) Which method has a String return type?	hunch
(c) Which method has an int parameter?	avatar
(d) Which method has a boolean parameter?	exaptation
(e) What is the constructor name?	adjacent

10. Write the 4 method signatures that would result from this structure chart. /8

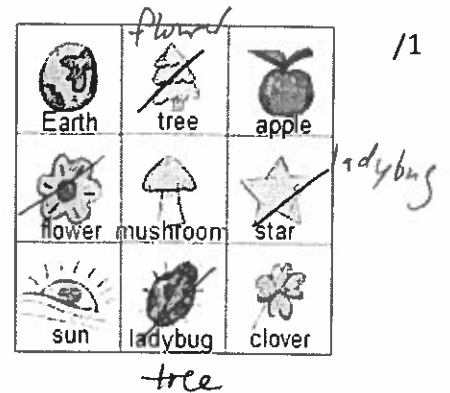
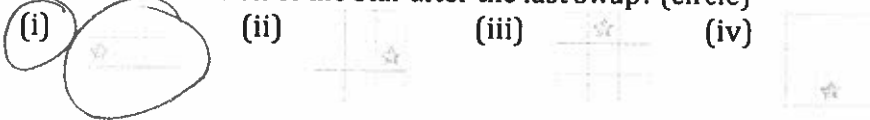
- public char artist ()
Method start word return type (output) method name parameter type parameter name (input)
- public int adwriter (String name)
Method start word return type (output) method name parameter type parameter name (input)
- public void coder (double price)
Method start word return type (output) method name parameter type parameter name (input)
- public void writer ()
Method start word return type (output) method name parameter type parameter name (input)



11. (a) A board is divided into squares and a different object is placed in each square as shown. The swap method exchanges the locations of two objects. The swap method is called 3 times in this order:

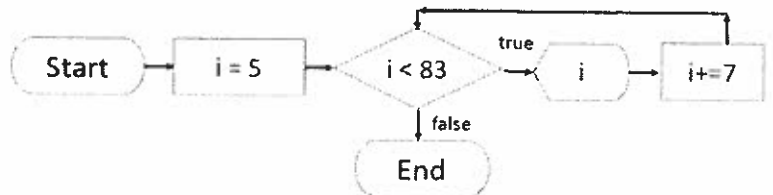
```
swap (flower, tree);
swap (tree, ladybug);
swap (ladybug, star);
```

What is the location of the star after the last swap? (circle)



12. Write for loop code that would result from this flow chart. /3

```
for (int i=5; i<83; i+=7)
    s.o.p (i + " ");
```



Application

don't forget { }

13. Add a sandwich loop in this code to make the program run until the user wishes to quit.

/5

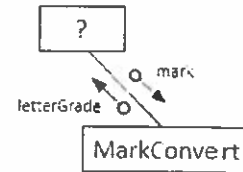
```
char again = 'y';
while (again != 'n') {
    int n = (int) (Math.random () * 100 + 1);
    System.out.println ("There are " + n + " happy bunnies. ");
    again = IO.inputChar ("Agreed? (y/n)");
}
```

14. Write a method that takes an integer and returns a char.

/9

The char should be the A, B, C, D or F that corresponds with the mark. ~~Nothing should be printed out in the method, return the char instead.~~

Letter	A	B	C	D	F
Mark	80+	70-79	60-69	50-59	49-0



```
public char MarkConvert (int mark) {
    if (mark >= 80)
        return 'A';
    else if (mark >= 70)
        return 'B';
    else if (mark >= 60)
        return 'C';
    else if (mark >= 50)
        return 'D';
    else
        return 'F';
}
```

Annotations: 'method start word', 'return type (output)', 'method name (see structure chart)', 'parameter type', 'parameter name (input)'. Handwritten note: 'No System.out. Return instead.'

15. Create a right-angled triangle on the screen using loops based on the size entered.

/8

Three different runs of the program are shown. The user could enter any size.

How big? <u>-1</u> Can't be drawn.	How big? <u>4</u> 4444 333 22 1	How big? <u>7</u> 7777777 6666666 55555 4444 333 22 1
---------------------------------------	---------------------------------------------	----------------------------------------------------------------------------

```
//Ask the question, get input
int n = IO.inputInt ("How big?");

//If it can't be drawn say so
if (n <= 0)
    System.out.println ("Can't be drawn");

//Otherwise, draw it.
else {
    for (int i = n; i > 0; i--)
        for (int j = 0; j < i; j++)
            System.out.print (i + " ");
        System.out.println ();
}
```