

PRACTICE FOR FINAL

CMPS 12a - Spring 02
Thomas Raffill

Name: _____
Student ID: _____

This exam is closed book, closed notes, no electronic devices. Show all work. Partial credit given for partial solutions. Presentation counts! Be legible and coherent for full credit.

Question 1: _____ (out of 5)
Question 2: _____ (out of 5)
Question 3: _____ (out of 5)
Question 4: _____ (out of 5)
Question 5: _____ (out of 5)
Question 6: _____ (out of 5)
Question 7: _____ (out of 5)
Question 8: _____ (out of 5)
Question 9: _____ (out of 5)
Question 10: _____ (out of 5)
Question 11: _____ (out of 5)
Question 12: _____ (out of 5)
Question 13: _____ (out of 5)
Question 14: _____ (out of 10)
Question 15: _____ (out of 10)
Question 16: _____ (out of 10)
Question 17: _____ (out of 20)

Total: _____ (out of 115)
(Anything above 100 counts for extra credit)

Name: _____

(15 points) **TRUE OR FALSE:** Give the best answer, true or false, to the following statements.

1. (5 points) A large function that does many things is a better style than many small functions that each do only one thing each.
2. (5 points) It is a good style to make instance variables of a class private and control access to them from other classes by using get and set methods.
3. (5 points) When debugging your code, you should never comment out part of the program to focus on only one unit.

(15 points) **MULTIPLE CHOICE:** Pick the best answer from the following choices.

4. (5 points) Which boolean expression is equivalent to the expression:

$!(x == 5 \ || \ (x + 1) \% 7 == 3)$

- (a) $(x != 5 \ || \ x \% 7 != 2)$
- (b) $(x != 5 \ \&\& \ x \% 7 != 2)$
- (c) $(x != 5 \ || \ x \% 6 == 3)$

5. (5 points) Which of the following is a valid Java identifier?

- (a) `_aWeirdFunctionName567`
- (b) `99Bottles`
- (c) `price/Gallon`

6. (5 points) Which of the following is a valid Java literal?

- (a) `'\n'`
- (b) `"A ROSE" + "IS A ROSE"`
- (c) `{ 1, 2, 3, 4, 5}`

Name: _____
(15 points) **EVALUATING EXPRESSIONS**
Give the values of the following expressions.

7. (5 points)

```
1 + 2 + 3 + " Hello " + 1 + 2 + 3
```

8. (5 points)

```
'a' + 'f' - 'c'
```

9. (5 points) (Assume x is an int whose value is 4)

```
(double)++a/2;
```

10. (5 points)

```
((int a = 5) <= 23 % 18)
```

Name: _____

(15 points) **RECOGNIZING EXCEPTIONS**

Identify the types of Exceptions in the following code fragments.

11. (5 points)

```
int[] anArray;  
intArray[0] = 111;
```

12. (5 points)

```
Integer.parseInt("ABC");
```

13. (5 points)

```
int[] oddNumbers = { 1, 3, 5, 7, 9 };  
oddNumbers[5] = 11; // The sixth odd number.
```

Name: _____

14. (10 points)

The formula to convert a Fahrenheit value to a Celcius value is $C = (5/9)(F - 32)$, where F is degrees Fahrenheit and C is degrees Celcius. Write a public static function called `fahrenheitToCelcius` with a double parameter and returning a double and implements this formula.

15. (10 points)

You have a 3-dimensional array of double values called `temperatureData` containing the temperatures (in Fahrenheit) at points in the atmosphere over a certain region. Write a code fragment to step through this array and convert each element from Fahrenheit to Celcius by calling the `fahrenheitToCelcius` function you defined in the previous problem.

Name: _____

16. (10 points)
What does the following code print out?

```
public class ExamExercise
{
    private int funInteger = 5;

    public static void main(String[] args)
    {
        int funInteger = 3;
        ExamExercise theExercise = new ExamExercise();
        int result = theExercise.runFunction(funInteger);
        System.out.println("funInteger is " + funInteger);
        System.out.println("The result is " + result);
    }

    public int runFunction(int theParameter)
    {
        int funInteger;
        funInteger = theParameter*this.funInteger;
        return funInteger;
    }
}
```

Name: _____

17. (20 points)

Write a class called `ComplexNumber` with two private double instance variables called `RealPart` and `ImaginaryPart`, a public constructor taking two double parameters, public get and set methods for each instance variable, a static `add` method taking two `ComplexNumbers` and returning the sum, and an instance `addTo` method taking one `ComplexNumber` and adding it to the given instance.