

CSC 207H L0101 2011 Java Quiz 3

Duration — 25 minutes

Aids allowed: none

Student Number:

Last Name:

First Name:

(Please fill out the identification section above and read the instructions below.)

Good Luck!

This quiz consists of 2 questions on 6 pages (including this one). *When you receive the signal to start, please make sure that your copy is complete.*

If you use any space for rough work or need to scratch out an answer, circle what you want marked to indicate that it is the answer you are submitting.

1: _____/10

2: _____/ 9

TOTAL: _____/19

Question 1. [10 MARKS]

Don't write Javadoc for any of the code you write.

```
1 interface Edible {
2     public String[] getNutrients();
3     public void bite();
4 }
5
6 class Apple implements Edible {
7     public String[] getNutrients() {
8         // code not shown
9     }
10
11     public void bite() {
12         throw new ForbiddenFruitException("You can't eat this.");
13     }
14 }
15
16 class BrusselsSprouts {
17     // clearly not edible
18 }
```

The `instanceof` Java keyword can be used to test class membership or interface implementation. For example, `obj instanceof Edible` evaluates to `true` if `obj` refers to an object that has an `Edible` part, where `Edible` is either a class or an interface.

Suppose we want to write a class `Animal` that has one public instance method with signature `eat(Object)`. This method should throw an `InedibleException` exception if the argument given is not an `Edible` object, and otherwise call the object's `bite` method.

Part (a) [3 MARKS]

Define the `InedibleException` class. Use an appropriate superclass, and make sure that it can be instantiated both with and without an error message.

Part (b) [3 MARKS] Write the `Animal` class.

Part (c) [4 MARKS]

Write JUnit test methods (not the whole class unless you want to) that tests whether `Animal`'s `eat` method throws exceptions appropriately when given `BrusselsSprouts` and `Apple` objects as arguments.

What are you expecting to happen if an `Animal` tries to eat an `Apple`?

Question 2. [9 MARKS]**Part (a)** [3 MARKS]

Write a regular expression that matches any integer, including leading zeroes, and groups the nonnegative part. For example, your regex should match -0183 and the first group should be 0183.

Part (b) [3 MARKS]

Give an example of two strings of different lengths matched by the regular expression $(416)-?\{d\}3-\{d\}+$

Part (c) [3 MARKS]

Normally, $*$ and $+$ are *greedy*: they match as many characters as possible. To make them *reluctant*, use $?$.

As an example, the regular expression $(a+)(a+)$ would match the string `aaaa`, where group 1 is `aaa` and group 2 is `a`.

The regular expression $(a+?)(a+)$ would also match the string `aaaa`, but group 1 is `a` and group 2 is `aaa`.

Here is part of the text from *Green Eggs and Ham* by Dr. Seuss:

I am Sam

I am Sam

Sam I am

That Sam-I-am

That Sam-I-am!

I do not like

that Sam-I-am

Notice the repetition of “am” at the ends of words. Write a regular expression that matches lines that have two words that end with the same sequence of letters, where the second occurrence is the ending of the last word on the line. (Don’t just match “am”: this should work for any such poetry.)

This page is for rough work and for answers that didn't fit in the space provided.

Here are some predefined character classes:

.	Any character
\d	A digit: [0-9]
\D	A non-digit: [^0-9]
\s	A whitespace character: [\t\n\x0B\f\r]
\S	A non-whitespace character: [^\s]
\w	A word character: [a-zA-Z_0-9]
\W	A non-word character: [^\w]
\b	A word boundary: any change from \w to \W or \W to \w

Here are some quantifiers:

Quantifier	Meaning
X?	X, once or not at all
X*	X, zero or more times
X+	X, one or more times
X{n}	X, exactly n times
X{n,}	X, at least n times
X{n,m}	X, at least n; not more than m times

By default, X* and X+ are greedy: match as many characters as possible. If you append a ? to X* or X+, the quantifier is reluctant: match as few characters as possible.