

COS 126	General Computer Science	Spring 2011
<b>Written Exam 2</b>		

This test has 10 questions worth a total of 50 points. You have 50 minutes. The exam is closed book, except that you are allowed to use a one page cheatsheet. No calculators or other electronic devices are permitted. Give your answers and show your work in the space provided. **Write out and sign the Honor Code pledge before turning in the test.**

*“I pledge my honor that I have not violated the Honor Code during this examination.”*

Name:

-----  
Signature

NetID:

Problem	Score
0	
1	
2	
3	
4	
Sub 1	

Problem	Score
5	
6	
7	
8	
9	
Sub 2	

- P01 TTh 1:30 Keith
- P01A TTh 1:30 Doug
- P01B TTh 1:30 Victor
- P01C TTh 1:30 Richard
- P01D TTh 1:30 Gordon
- P01E TTh 1:30 Arman
- P02 TTh 2:30 Doug
- P03 TTh 3:30 Gordon
- P03A TTh 3:30 Keith
- P04 TTh 7:30 Nick
- P05 WF 10 Dmitry
- P06 WF 1:30 Victor
- P06A WF 1:30 Chris
- P06B WF 1:30 Donna
- P07 WF 12:30 Donna

**Do not remove this exam from the room.**

## 0. Miscellaneous. (1 point)

- (a) Write your name and Princeton NetID in the space provided on the front of the exam, and circle your precept number.
- (b) Write and sign the honor code on the front of the exam.

## 1. Data types. (7 points)

- (a) Define what is a *data type*.

A set of \_\_\_\_\_ and \_\_\_\_\_

on those \_\_\_\_\_.

- (b) Suppose that the following four variables are initialized in the four statements below.

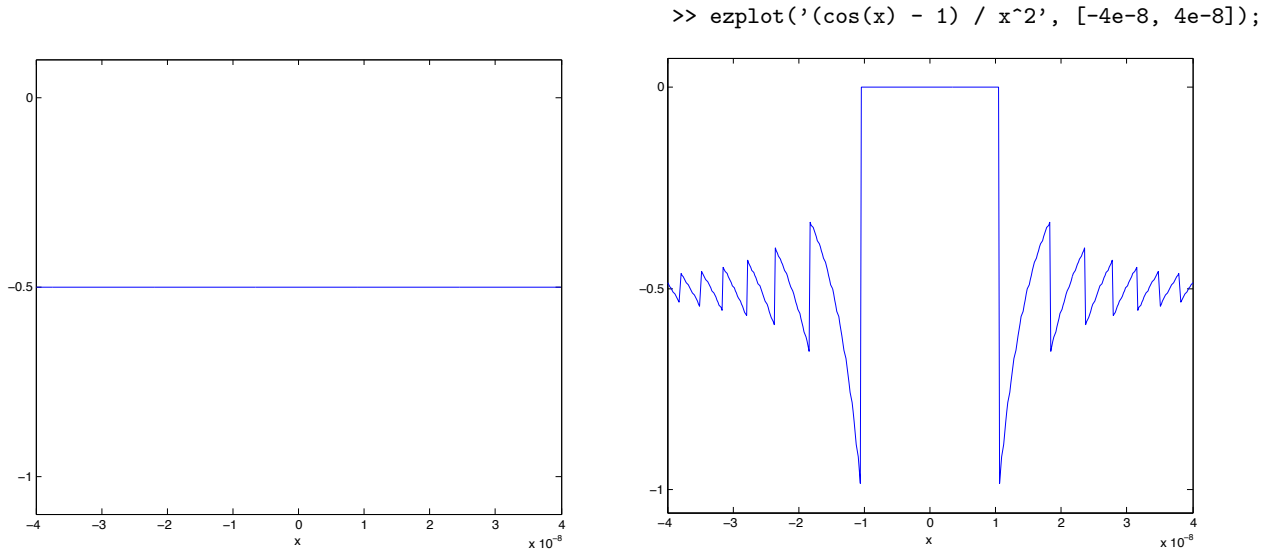
```
String r = "Hello";
String s = "World";
String t = r + " " + s;
String u = "Hello World";
```

Give the type and value of each Java expression below. If it leads to a compile-time or runtime-error, specify that for the type (and leave the value column blank).

<i>Java expression</i>	<i>type</i>	<i>value</i>
<code>r.length()</code>		
<code>r.charAt(r.length())</code>		
<code>(t == u)</code>		
<code>u.equals(t)</code>		
<code>r.substring(0, r.length())</code>		

2. Scientific computation. (4 points)

- (a) The left plot below shows the function  $f(x) = (\cos x - 1) / x^2$  in the interval  $-4 \cdot 10^{-8} \leq x \leq 4 \cdot 10^{-8}$ . The right plot below shows the result of attempting to plot  $f(x)$  with the Matlab command `ezplot`.



Which of the following best explains why the Matlab plot of  $f(x)$  is so inaccurate?  
 Circle the best answer.

- i. Matlab is not using enough sample points and/or the samples are not taken uniformly.
  - ii. Matlab uses the IEEE 754 floating-point standard and the algorithm for computing  $f(x)$  is *unstable* in the specified interval because of catastrophic cancellation.
  - iii. Matlab uses the IEEE 754 floating-point standard and the function  $f(x)$  is *ill-conditioned* in the specified interval because of catastrophic cancellation.
- (b) Which of the following numbers are exactly representable as a Java `double`?  
 Circle *all* such numbers.

$1/5$        $1/2$        $3/4$        $1$        $\pi$        $123$



**4. Data type design. (6 points)**

- (a) Which of the following help *enforce immutability* in Java? Write the letter Y next to each description that helps enforce immutability and N if it does not.

- \_\_\_ declaring instance variables to be **private**
- \_\_\_ declaring instance variables to be **immutable**
- \_\_\_ declaring instance variables to be **final**
- \_\_\_ defensively copying instance variables
- \_\_\_ overloading instance methods

- (b) Suppose that a client passes a reference to an object to a static method and that object's value after the function call is *different from* its value before the call. Which of the following data types that you've encountered in this course could be the type of the object? Circle one or more answers.

`double []`

`String`

`Stack<String>`

`Tour`

`Complex`

`GuitarString`

## 5. Analysis of algorithms. (6 points)

Given an integer array `a[]`, the following method counts the number of distinct integers.

```
public static int distinct(int[] a) {
    int N = a.length;
    int count = 0;
    for (int i = 0; i < N; i++) {
        boolean distinct = true;
        for (int j = i+1; j < N; j++) {
            if (a[i] == a[j]) distinct = false;
        }
        if (distinct) count++;
    }
    return count;
}
```

- (a) What is the order of growth of the running time of `distinct()` as a function of  $N$ ? Circle the best answer.

$N$        $N \log N$        $N^2$        $N^3$        $2^N$        $N!$

- (b) Suppose that `distinct()` takes 15 seconds to process an array of size 100,000. Estimate how long (in seconds) it will take to process an array of size 400,000.

- (c) Suppose that you declare the following two-dimensional array in Java:

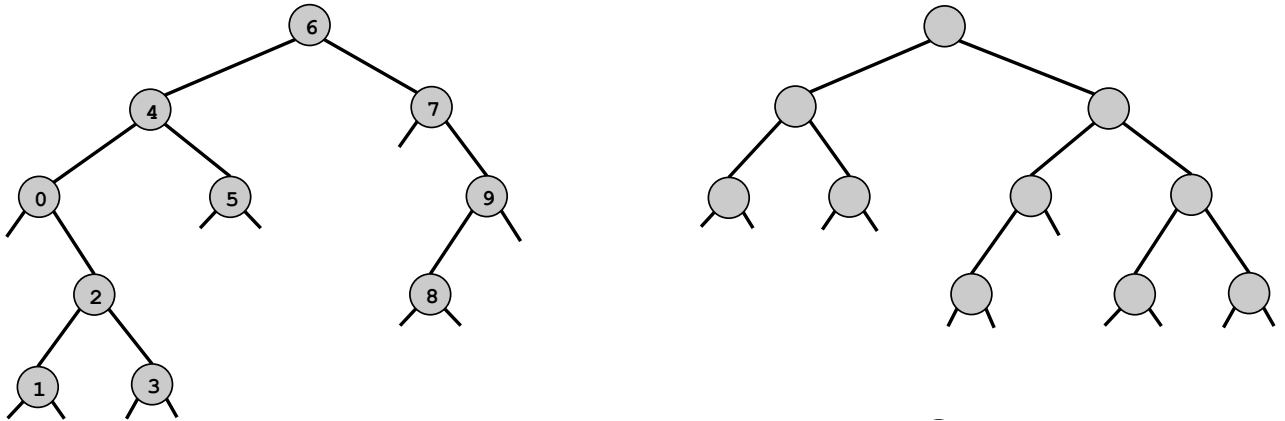
```
double[][] a = new double[N][N];
```

How much memory *in bytes* does `a[][]` consume as a function of  $N$ ? Circle the best answer. *Hint: a Java double is 64 bits.*

$\sim 4N$      $\sim 8N$      $\sim 32N$      $\sim 64N$      $\sim 4N^2$      $\sim 8N^2$      $\sim 32N^2$      $\sim 64N^2$

6. **Symbol tables. (4 points)**

Consider the binary search tree at left. Put the same set of keys into the binary tree at right so that it is a binary search tree.



7. **Regular expressions. (4 points)**

A *camelCase string* over the alphabet { a, b, A, B } is characterized by:

- starts with a lowercase letter
- no two consecutive uppercase letters
- ends with a lowercase letter

<i>matches</i>	<i>does not match</i>
aaaa	Baaaa
aaaBa	aaaaB
aBaaAaa	ABABBBA
baBabAaAbAbb	aBAAaba
b	B
bb	ε

Which regular expression below matches *camelCase* strings? Circle the best answer.

- $(a|b) (.*) (a|b)^*$
- $(a|b) (a|b)^* ((A|B) (a|b) (a|b)^*)^*$
- $((a|b)^* (A|B) (a|b))^*$
- $(a|b) ((a|b)^* (A|B) (a|b))^* (a|b)$

## 8. Theory of computation. (8 points)

- (a) For each statement on the left, pick the best matching description on the right. You may use each letter any number of times.

- |  |   |
|--|---|
| --- There exists a mathematical function that can be computed in Java, but <i>cannot</i> be computed on a Turing machine.  | A. known to be true   |
| --- There exists a mathematical function that can be computed in polynomial time on a quantum computer, but <i>cannot</i> be computed in polynomial time on a Turing machine. <i>Assume that quantum computers can be built.</i> | B. known to be false  |
| --- There exists a mathematical function that can be computed in polynomial time in Java, but <i>cannot</i> be computed in polynomial time on a Turing machine.  | C. if true would falsify the Church-Turing thesis                 |
| --- There exists a Universal Turing machine that can simulate the behavior of any other Turing machine.  | D. if true would falsify the <i>extended</i> Church-Turing thesis |
|  | E. if true would prove the Church-Turing thesis                   |

- (b) For each statement on the left, pick the best matching description on the right. You may use each letter any number of times.

- |  |  |
|--|--|
| --- Not all search problems can be solved in polynomial time.            | A. known to be true                                      |
| --- There exists a search problem that can be solved in polynomial time. | B. known to be false                                     |
| --- Both FACTOR and 3-SAT can be solved in polynomial time.              | C. if true would imply $P = NP$                          |
| --- Exactly one of 3-SAT and TSP can be solved in polynomial time.       | D. if true would imply $P \neq NP$                       |
|  | E. if true would prove the extended Church-Turing thesis |



9. Circuits. (5 points)

(a) Fill in the truth table for a 3-bit palindrome  $XYZ$  (reads same forwards and backwards).

$X$	$Y$	$Z$	$f$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

(b) Write the (unsimplified) sum-of-products boolean formula for 3-bit palindromes.

(c) As a function of  $N$ , how many  $N$ -input *AND gates* are in the (unsimplified) sum-of-products circuit for an  $N$ -bit palindrome. Assume that  $N$  is an odd integer and  $N \geq 3$ .