The University of Western Ontario

Computer Science 1026b, Spring 2010
Computer Science Fundamentals I
Final Exam (Alternate)
2:00pm - 5:00pm, Friday, April 16, 2010.

Name (last, first): _____

Student Number: _____

Instructions

1. TURN OFF YOUR CELL PHONE AND ALL ELECTRONIC DEVICES.
2. Do not turn this page until instructed to do so.
3. Cheating (e.g. copying someone else's solutions, using prohibited material like your textbook and notes, or using electrical communication devices) will result in a zero grade and possible expulsion from the course.We reserve the right to orally examine you regarding your solutions and then regrade accordingly.
4. For each question, provide your solution in the answer box. **All writing outside boxes will be ignored** by the markers.
5. The exam document consists of eighteen (18) pages. No other paper document than the exam document is permitted during the exam.
6. **The entire exam document**, including the sheets provided to you for rough work, **must be returned** to the proctoring personnel before leaving the exam room. Failing to do will result in a zero grade.
7. Only those exams written in pen are eligible for regrading.
8. Correct java code is preferable but pseudo-code (i.e. something resembling code) is acceptable and if appropriate will be accepted for marks.

§1	20	
§2	20	
§3	40	
§4	20	
Total	100	

§1 Very Short Answers

The Java code in the following questions is valid and will provide output (without error) if entered into the DrJava interactions pane.

Question 1

Each answer in this Question is worth *1 point* for a total of *10 points*.

a. After the following code segment is executed

```
int x = 10 + 2*5 % 10
```

the value of x is:

b. The number of "*"s printed by

```
for (int star = 9; star < 0; star++) {
    System.out.println("*");
}
```

is:

c. Suppose we have defined

```
int a = 12, b =0;
boolean t = true;
```

In order to make the expression `(t ?  !(a > b)) && t` evaluate to `true`, we should replace `?` with:

d. Suppose we have defined

```
int m=18, n=4;
```

then the value of `m / n + (m % n)` is:

e. After the following code segment is executed

```
int x = 3;
x = x * x;
x = x + x;
```

the value of `x` is:

f. Suppose we have defined

```
String s1 = new String("Finals");
String s2 = new String("Finals");
```

then the value of the expression `(s1==s2)` is:

g. The *last* value of the array `int[] xs =` {1,2,3,4} is at position:

h. After the following code segment is executed

```
int x;
for (x = 1; x < 5; x = x + x);
```

the value of `x` is:

i. After the following code segment is executed

```
int num = 15;
if (num >= 20)
    num = num - 20;
if (num >= 10)
    num = num -10;
if (num >= 5)
    num = num - 5;
```

the value of `num` is:

j. After the following code segment is executed

```
System.out.println("3"+"7");
```

what is printed?

Question 2

Each answer in this Question is worth *2 points* for a total of *10 points*.

a. For 32 to be printed by the following code segment

```
int total = x;
for (int i = 0; i <= 10 ; i = i + 2)
{
  total += i;
}
System.out.print(total);
```

the value of `x` should be:

b. The shape drawn by the following `Turtle` class method

```
void SecretShape () {
    for (int i =0; i < 13; i++) {
        this.forward(30);
        this.turnLeft();
    }
}
```

is a:

c. After the following code segment is run

```
int z = 0;
double x = 123./321, y = 0.38317757009345793;
if ( x <= y && x > y ) {
    z = 1;
}
```

the value of `z` is:

d. After the following code segment is run

```
int sum = 0;
for (int i = 0; i < 10; i ++) {
    sum += i ;
    i ++;
}
```

the value of `sum` is:

e. After the following code segment is run

```
int x = 2*2*2*3*3*3*7;
while ( (x % 2 == 0) || (x % 3 == 0 ) ) {
    if (x % 2 == 0) {
        x = x/2;
    } else {
        x = x/3;
    }
}
```

the value of `x` is:

§2 Short Answers

Each solution in this section is worth *4 points* for a total of *20 points*.

Question 3

Write a java class method `int MaxElement (int[] xs)` that returns the *maximum* (or *largest*) element of the array `xs`. For instance, if we had the array `A = {-3,4,10,-32,0}` then `MaxElement(A)` would return `10`.

Question 4

Write a java expression that is equivalent to the algebraic expression:

$$\frac{\frac{c-(a+b)}{d} \times e}{f}.$$

You may assume that the initialization

```
double a,b,c,d,e,f;
```

has been made.

Question 5

Write a `for` loop that, given an array `int[] xs`, prints the elements of the array backwards. You may assume that an array has already been assigned to the variable `xs`

For example, given `int xs = 1,2,3,4;` the `for` loop would print `4321`.

Question 6

Write a java class method `void AnnoyUser()` that repeatedly prompts the user for input until the string "`banana`" is entered. (Hint: use `SimpleInput.getString` and a `while` loop).

Question 7

There are four syntax errors in the following code segment:

```
/00/  class Foo {
/01/      public void FooBar ( ) {
/02/           int xs = {1,2,3,4};
/03/
/04/           for ( i=0; i <= 4; i+=2 ) {
/05/               System.out.println( "The value of i is", i );
/06/           }
/07/
/08/          return  xs[0];
/09/       }
/10/  }
```

For each error, indicate the line where the error is and correct the error (for instance you could write something like: "Line 15 should be `double x = 2.5`").

§3 Long Answers

Each solution in this section is worth *10 points* for a total of *40 points*.

Question 8

a. Write an algorithm in pseudo-code (i.e. code mixed with English) which prints the following pattern when given `n`. Your method should work for any value of n and not only on the examples given below. Note that for a given `n`, the printed pattern has `n` rows and `n` columns.

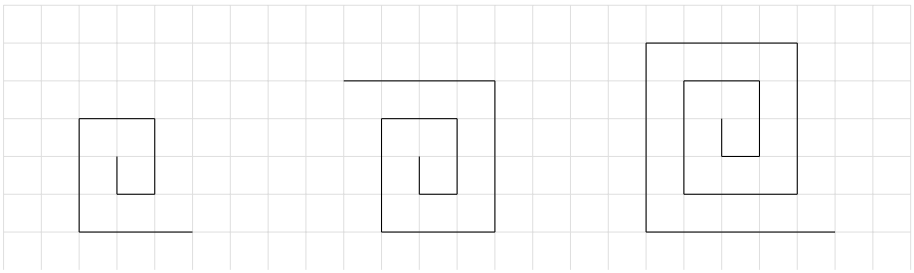
$n = 3$	$n = 4$	$n = 5$
`..1`	`...1`	`....1`
`.22`	`..22`	`...22`
`333`	`.333`	`..333`
	`4444`	`.4444`
		`55555`

b. Write a java class method `static void PrintPattern (int n)` that prints the same pattern as in Part a.

Question 9

Write a java object method `void StairCase (int n, int s)` for inclusion in the `Turtle` class that takes as input n and s and draws a spiral with $2 \times n$ lines on a grid with sqaures size $s \times s$. This is illustrated for $n = 3$, $n = 4$ and $n = 5$ below (but remember your code should work for *any* value of n).

You may assume that drawing this spiral will never cause your turtle to go out of its world's boundaries.



$n = 3$ $n = 4$ $n = 5$

Question 10

Write a java class method `int prod (int n)` that finds the product of the first n positive integers (i.e. $1 \times 2 \times 3 \times \cdots \times n$.) For example `prod(3)` returns 6 because $1 \times 2 \times 3 = 6$ and `prod(5)` returns 120 because $1 \times 2 \times 3 \times 4 \times 5 = 120$. Your code should work for these examples *and* any other positive integer n.

Question 11

Determining if a `Pixel` object is white is relatively straightforward: if a `Pixel` object's `red`, `green`, and `blue` components are all 255, then the `Pixel` object is white.

a. Write a java method: `boolean isWhitePixel (Pixel P)`, for inclusion in the `Picture` class that tests if a pixel `P` is white.

b. Using your method from Part a., write a java object method: `int countWhitePixels()`, for inclusion in the `Picture` class, that counts the number of white pixels in a picture. This method should take no parameters and return an integer that is the number of white pixels in the picture.

c. Suppose `Picture Sunset = new Picture(...)` has been properly initialized. What would you type in the interactions pane to determine the number of white pixels in `Sunset` (you may assume all classes have loaded properly).

§4 Very Long Answer

There is one question in this section with five parts. Each part is worth *4 points* for a total of *20 pionts*.

Question 12

For this question you will create a new class that represents information about students in a university course. Use the following code as a starting point:

```
public class Course {
    private String instructor;   //the name of the instructor
    private String course;       //the name of the course
    private Student[] students;  //an array for the students in the course
}
```

Specifications for `Student` class is part of the extra exam material (i.e. the "cheat sheet").

Part i.

Write a constructor method for inclusion in the `Course` class with the header:

```
public Course (String instructorName, String courseName, int maxClassSize)
```

that initializes the course in the obvious way (`maxClassSize` is used to initialize the `students` array).

Part ii.

Write three methods for inclusion in the `Course` class with headers:

1. `public String getInstructor()`
2. `public String getCourse()`
3. `public int getMaxClassSize()`

which return (respectively) the name of the instructor, the name of the course, and the maximum class size.

Part iii.

Write a method for inclusion in the `Course` class with the header

```
public boolean enrollStudent(Student theStudent)
```

that *adds* a students to the course. More precisely, this method returns `false` if there is no room for the student in the course; otherwise it puts `theStudent` in the `students` array and returns `true`.

Part iv.

Write a method for inclusion in the `Course` class with the header

```
public boolean removeStudent(Student theStudent)
```

that *removes* a student from the course. Return `true` if successful and `false` otherwise. (Removing a student that is not in the course should be considered a failure).

Part v.

Write a method for inclusion in the `Course` class with the header

```
public int getNumberOfEnrolledStudents()
```

that returns the number of enrolled students in the course.