

Faculty of Arts & Sciences Department of Computer Science CMPS 200—Introduction to Computer Programming Fall 2005–2006 Saturday, November 19, 2005

Exam 1

Version 1

Name:	Student Id	•		
Signature:	Section:	Lect I	10–11	
		Lect II	1–2	
Answers to Part I	Gra	ides		

Question	A	В	С	D	E
1.					
2.					
3.					
4.					
5.					
6.					
7.					
8.					
9.					
10.					
11.					
12.					
13.					
14.					
15.					
16.					
17.					
18.					
19.					
20.					
21.					
22.					
23.					
24.					
25.					

Grades		
Part I	50	
Correct		
Incorrect		
Blank		
Part II	53	
1.1.1	2	
1.1.2	3	
1.1.3	3	
1.1.4	2	
1.1.5	2	
1.2.1	2	
1.2.2	3	
1.2.3	8	
1.2.4	2	
1.2.5	2	
1.2.6	2	
1.2.7	2	
1.3	10	
2	12	
Total	105	

Part II

Answer the following questions in the space provided.

Problem 1

The XYZ distribution company is testing a new technology for shipping boxes between its distribution centers. The technology consists of stacking shipping boxes along their shortest dimension in the container until it fills up before shipping the container itself.

As an employee in XYZ's IT department, the company has asked you to develop a Java application to collect statistics on the efficiency of the new shipping technology. This application consists of three classes: ShippingBox, ShippingContainer, and ShippingStats.

- Instances of ShippingBox represent the boxes to be shipped which come in different sizes.
- The application has one instance of ShippingContainer which represents the container to be shipped to another distribution center. Containers have standard heights and widths $(2m \times 2m)$, but differ in their lengths.
- The class ShippingStats is the driver class of the application.

Complete the implementation of these classes below.

1.		ShippingBox has three data members, <i>length</i> (of type float), <i>width</i> (of type float), and type float). Also, complete the constructor(s) and other methods whose specifications appear ats.
pu	blic cla	ss ShippingBox {
{	// 1.1.1 (2	2 points) Instance variable declarations
	•••••	
	•••••	
	// 1.1.2 (3	3 points) Constructor:
	//	Takes three parameters of type float. The constructor uses the ternary conditional operator to make sure that the parameters are positive numbers. Illegal values default to 10.0.
	•••••	

CMPS 200 9 of 16

// 1.1.3	(3 points) Method getShortestDimension:
//	Takes no parameters and returns a float which gives the stacking dimension of the
//	shipping box. Note: the shipping box will be stacked along its shortest dimension.
, ,	
•••••	
•••••	
•••••	
•••••	
// 1.1.4	(2 points) Method getVolume:
//	Takes no parameters and returns a float which gives the volume of the shipping box.
•••••	
•••••	
•••••	
/ / 1.1.5	(2 points) Method toString:
//	Returns the three properties of a ShippingBox object as a String object with these values
//	separated by commas.
/ /	separated by commas.
•••••	
•••••	
•••••	

CMPS 200 10 of 16

}

2.	The class ShippingContainer has three data members, <i>length</i> (of type int), <i>stackedLength</i> (of type float), and <i>stackedVolume</i> (of type float). Also, complete the constructor(s) and other methods who specifications appear in comments.	-
pul {	lic class ShippingBox {	
•	// 1.2.1 (2 points) Instance variable declarations	
		•
	// 1.2.2 (3 points) Constructor:	•
	// Takes one parameter of type int. The constructor ensures that the parameter is greater than zero. An illegal value defaults to 10. The constructor should also initialize the container's other instance variables.	
		•
	// 1.2.3 (8 points) <i>Method</i> stackBox:	•
	Takes one parameter: <i>box</i> of type ShippingBox and returns a boolean. The method returns true if there is room in the length of the container to further stack the box parameter along its shortest dimension, false otherwise. When the box is successfully stacked, the method updates the object's stackedLength and stackedVolume instance variables.	.
		•
		•

CMPS 200 11 of 16

•••••	
// 1.2.4	(2 points) Method getStackingRatio:
//	Takes no parameters and returns a float indicating the ratio of the container that is filled up
//	by length (the ratio of the container's stackedLength and length).
•••••	
	5 (2 points) Method getStackingEfficiency:
/	Takes no parameters and returns a float giving the stacking efficiency of the container (the ratio of stackedVolume and the container's volume—its length \times cross section).
7 7	(ine ratio of peacetrea volume and the container by orange his length wereas section).
•••••	
•••••	
// 1.2. 0	6 (2 points) <i>Method</i> toString: Returns the properties of a ShippingBox object, <i>length</i> , <i>stacking length</i> , and <i>stacking</i>
//	volume, as a String object with these values separated by commas.
•••••	

Initials:

Exam 1 (2005-11-19)

}

CMPS 200 12 of 16

3. **1.3** (10 points) Write a driver application, ShippingStats, that uses the classes ShippingContainer and ShippingBox to do the following:

- Prompt the user to enter the length of a shipping container which should be longer than 2.0m. Repeat the prompt until the user enters a number that is greater than 2.0m.
- Create one ShippingContainer object.
- Repeatedly prompt the user for the dimensions of a ShippingBox and stack the box into the container until the container fails to accept the box.
- Print a report as to the container's efficiency and exit.

pul {	olic class ShippingStats {
· ·	<pre>public static void main(String[] args) {</pre>

CMPS 200 13 of 16

•••••		 		 	
•••••		 		 	
•••••		 		 	
•••••					
•••••	••••••	 •••••••	•••••	 •••••	
•••••	•••••	 		 	
•••••		 		 	
•••••		 		 	
}					

Initials:

Exam 1 (2005-11-19)

CMPS 200 14 of 16

Problem 2

Implement the method main of the Java application, RightTriangle, which reads a positive integer between 2 and 9 from the user and prints a right-handed triangle made of digit characters such that:

- 1. The width of the triangle's base is equal to the number read.
- 2. The first line consists of blank spaces and 1s; the second line consists of blanks and 2s; etc.

The output for a triangle whose base is 8 should look as follows:

7	1 22 333 4444 55555 666666 777777	
pul {	blic static void main(String[] args)	

CMPS 200 15 of 16

m 1 (2005-11-19)	Initials:

CMPS 200 16 of 16

}