The University of British Columbia

CPSC 210

Sample Midterm Examination

Time: 80 minutes

Name_____

(PRINT) (Last) (First)

Student No_____ Signature_____

This examination has 8 pages.

Check that you have a complete paper.

This is an open computer examination. You may use any reference on the internet but you must not collaborate with any other individual through personal contact or electronic contact in anyway during the exam.

Answer all the questions on this paper.

Give **short but precise** answers.

Work fast and do the easy questions first. Leave some time to review your exam at the end.

The marks for each question are given in []. Use this to manage your time.

Good Luck

Question	**Marks**
1	/45
2	/14
3	/11
4	omitted
TOTAL	**/80**

Question 1.Object-Oriented Design: Ticket System [Total 45 points]

NOTE: This question is the longest on the exam. Manage your time on it carefully!

a. (30 points) Read the system requirements below. Produce an object-oriented software design using the responsibility-driven design approach we covered in the course. For each class you identify:

   i. State the name of the class.
   ii. State a 1-2 sentence description of the purpose of the class.
   iii. List major responsibilities (operations) for the class.

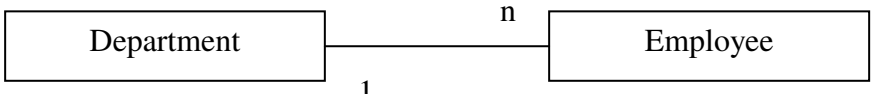
We will give you separate paper on which to record and hand-in this design. Please label each sheet with your name and student #.

A software system is needed to support sales of tickets and auctions of tickets to events. A user must register to use the system. A registered user may search for an event of interest. When the user finds an event of interest, the system displays any tickets available for direct sale or auction for the event. The user can view on a displayed map the location of each ticket within the venue for the event. Once the user selects tickets available for direct sale, credit card payment is requested from the user. Once the credit card payment is processed, the tickets are assigned to the user on the system. The user can later access the tickets to print them or to place them up for auction. When a user places tickets for auction, the user sets a minimum price that the user will accept for the tickets. When a user sees tickets available for auction, the user can place a bid for the tickets that is above the minimum bid price set by the seller. The bidding user must provide credit card information. At 24 hours prior to the event, the system automatically accepts the highest bid for tickets, charges the user and assigns the tickets to the user.

b. (15 points) Draw a UML class diagram for the design of the ticket system you produced for part a. You only need to include the classes and how the classes (objects) are connected in this diagram. Include the multiplicity and directionality (if any) of connections in the diagram. You do not need to list the fields or methods in the UML diagram you draw.

Question 2. Implementing an Object-Oriented Design. [Total 14 points]

a. (10 points) Consider a software system that must track which employees work for each department. Here is a fragment of a UML class diagram describing a design for this system.



Complete the implementation below by adding the code needed to implement the association shown in the UML class diagram above. You can use a separate sheet if more space is needed.

```
class Department {









}

class Employee {









}
```

b. (4 points total) Choose a suitable data structure from the Java Collections Framework (JCF) for each of the following problems:

i. (2 points) You are writing a system to manage a hockey pool. For each participant in the pool, you must be able to track a team of players. What data structure will you use to represent the team of players? Why?

ii. (2 points) You are writing a system to manage photographs. Each photograph may have a keyword assigned to it. You want to be able to efficiently retrieve the photographs associated with each keyword. What data structure will you use to represent the keywords and the associated photographs? Why?

Question 3. GUI [Total 11 points]

Consider the code below. Assume the code has all of the appropriate import statements to compile and execute.

```
public class ButtonDemo extends JFrame {

   public ButtonDemo() {
        super("Button Sample");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JButton button = createButton();
        Container contentPane = getContentPane();
        contentPane.add(button, BorderLayout.SOUTH);
        setSize(300, 100);
        setVisible(true);
      }

   private JButton createButton() {
        JButton button = new JButton("Select Me");

        button.addActionListener( new ActionListener() {

            public void actionPerformed(ActionEvent actionEvent) {
                  System.out.println("I was selected.");
            }
        });

        button.addMouseListener( new MouseAdapter() {

            public void mousePressed(MouseEvent mouseEvent) {
               if (SwingUtilities.isLeftMouseButton(mouseEvent)) {
                  System.out.println("Left button pressed.");
               }
               else if(SwingUtilities.isRightMouseButton(mouseEvent)) {
                   System.out.println("Right button pressed.");
               }
            }

            public void mouseReleased(MouseEvent mouseEvent) {
               if (SwingUtilities.isLeftMouseButton(mouseEvent)) {
                   System.out.println("Left button released.");
               }
               else if (SwingUtilities.isRightMouseButton(mouseEvent)) {
                   System.out.println("Right button released.");
               }
            }
         });
         return button;
      }

   public static void main(String args[]) {
        ButtonDemo bd = new ButtonDemo();
    }.}
```

(a) (5 points) Suppose that we execute the main in the code shown on the previous page. In point form, describe which methods will be called and what will be printed to the screen when the button labeled "Select Me" is clicked with the left mouse button. (Note: You do not need to worry about the precise order in which events occur.)

(b) (6 points) Draw the containment hierarchy for the GUI.