

---

Name

Student ID

Signature

---

**Department of Electrical and Computer Engineering, McGill University**  
**ECSE 321 - Introduction to Software Engineering, Fall 2005**

**Final Examination**

**December 20, 2005, 14:00-17:00**

**Examination version II**

Examiner: Prof. Shie Mannor



Associate Examiner: Prof. Warren Gross



**INSTRUCTIONS**

1. Write your name, McGill student ID number, and signature in the top of this page. If you require an answer booklet, write your name there too.
2. This examination consists of two sections:
  - a. Section I consists of 14 multiple-choice questions.
  - b. Section II consists of 5 questions (mixed multiple choice and open).
3. Section I is worth 42 marks in total; section II is worth 58 marks in total. The multiple-choice questions are equally weighted; wrong or omitted answers are worth 0 marks.
4. This examination booklet has 16 pages, including this page. Keep all pages together.
5. Please use only the space allocated in the booklet.
6. Multiple choice questions: circle the correct option in multiple choice questions. Any ambiguity in the final answer will be considered invalid. If you change your mind, you may write "Final: A" for making sure we understand you think that A is the correct answer. Some questions require extra explanation. Make sure you explain your answer. In these questions the explanation is worth as much as the correct answer.
7. Several questions require a written explanation. Be concise and precise. Don't use more space than allocated to you.
8. You may use any written material you wish (books, notes, lecture notes, tutorial notes, etc.). You may not use a computer in this exam.

Good luck!

## Section I: Multiple choice questions

(3 marks for each question, total of 42 marks)

1. Which one of the following statements is TRUE?
  - A. Software development teams need to accept every requirement change proposed by the customer.
  - B. Defining requirements solely involves interviewing the customer and documenting what they want.
  - C. Risk management is about identifying all possible project risks and making sure they are mitigated.
  - D. Good software quality engineering can prevent all defects from happening, not merely finding them and making sure they are fixed.
  - E. None of the above
  
2. Which one of the following is a functional requirement?
  - A. The system should process at least 2000 requests per second.
  - B. The system will be developed in Java.
  - C. The developers will use CVS for version control.
  - D. The scrollbar will be dark blue.
  - E. None of the above.

Explain: (use allocated space only)

---

---

---

3. After completing the BetArena baby system successfully, the client decided that SMS enabled bets are the main source of revenues. You are hired to build an SMS-based system. The interface should be completely SMS-based including feedback from and to the end-user by SMS. End-user registration will be done over the phone and there will be no web interface. The customer is mostly concerned about the usability of the system and is willing to work with you on improving it. What kind of software life-cycle will you use?
  - A. Waterfall model.
  - B. Spiral model with two iterations.
  - C. Saw-tooth model.
  - D. Extreme programming model.
  - E. V-model

Explain: (use allocated space only)

---

---

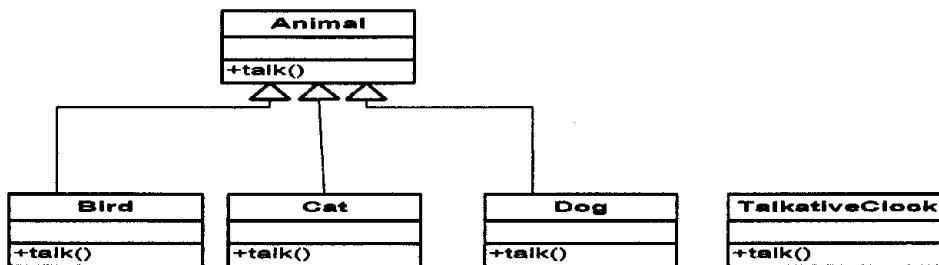
---

---

4. Which of the following is a benefit of the MVC architecture?

- A. The MVC pattern minimizes coupling.
- B. The MVC pattern provides abstraction.
- C. The MVC pattern provides reusability.
- D. The MVC pattern provides extensibility.
- E. None of the above.

5. It is a bad design to fit TalkativeClock into the Animal hierarchy classes. However an external class that is using those classes wants to be able to simulate the talk feature without caring about their types (TalkativeClock or Animal).



This best can be achieved by:

- A. Extending TalkativeClock from Animal.
- B. Extend Animal from TalkativeClock
- C. All classes should implement the Talkative Interface that has a Talk operation.
- D. There is no way for another class but to distinctly call TalkativeClock or Animal.
- E. Use the Bridge pattern to separates an object's interface from its implementation.

6. Which one of the following statements is TRUE regarding black box testing and white box testing?

- A. Any testing by users or other non-developers is called white box testing.
- B. All unit testing must be black box testing.
- C. Behavioral test design, which defines tests based on functional requirements, is often called black box.
- D. Usability testing concerns whether the software will actually provide value to users, and could be termed white box.
- E. Stack information, memory dump and assertions are black box techniques.

7. When a system is said to have 5 9's availability it means that it can afford to be unavailable for:

- A. About 1 second a day.
- B. About 50 minutes a year.
- C. About 10 hours a year.
- D. About 5 minutes a day
- E. About 5 seconds a day

8. A spreadsheet is program that manipulates numerical and string data in rows and columns of cells. The value in a cell can be calculated from a formula which can involve other cells. A value is recalculated automatically whenever a value on which it depends changes. Different cells may be displayed with different formats. What best describes the relationship between the Spreadsheet and Cell classes?

- A. Association
- B. Aggregation
- C. Composition
- D. Delegation
- E. Generalization

Explain: (use allocated space only)

9. Suppose you want to implement a drawing program that uses a custom graphics API for drawing shapes on the screen. For example, you may have a shape called 'Circle' that draws a circle. You also want to make your graphics API work on different platforms. Which design pattern is best suited for this situation?

- A. The Adapter pattern.
- B. The Bridge pattern.
- C. The Facade pattern.
- D. The Command pattern.
- E. The Strategy pattern.

Explain: (use allocated space only)

10. This question concerns the ARENA case study. What is the purpose of introducing the StatisticsVault class that replaced SimpleStatisticsVault (page 425)?

- A. Use the Façade design pattern.
- B. Simplify the interface.
- C. Enable additional functionality.
- D. Make the system more decoupled.
- E. Make the system more cohesive.

Explain: (use allocated space only)

---

---

---

---

11. Which of the following is true?

- A. We usually choose not to code invariants because they cannot discover faults.
- B. It is a good practice to specify preconditions in the callee.
- C. All contracts must be coded for a software system to be reliable.
- D. Contracts are not part of UML.
- E. Inherited contracts cannot be weaker.

12. A software system was fixed after a bug was discovered. As part of the fix, one of the parameters in one of the functions of a low layer was changed. The fixed system was sent back to the client where a new bug was discovered as a result of a function call to the function that was fixed from another place in the system. This bug could have been prevented by:

- A. Rigorous unit testing of the function that was fixed.
- B. Testing the function with the new driver that uses the right parameters.
- C. Bottom-up integration.
- D. Top-bottom integration.
- E. Regression testing.

13. The Iterator Interface has three methods:

```
public boolean hasNext()  
public Object next()  
public void remove()
```

What is the output of the following code?

```
import edu.colorado.nodes.Node;  
import edu.colorado.nodes.Lister;  
  
...  
Node n1;  
Node n2;  
Node n3;  
Lister print;  
  
n1 = new Node("Intro", null);  
n2 = new Node("to", n1);  
n3 = new Node("Software", n2);  
  
print = new Lister(n3);  
while(print.hasNext())  
    System.out.println(print.next());
```

A. Software to Intro	C. Intro To
B. Software Intro	D. Intro to Software

14. Suppose two (or more) threads each run a series of operations repeatedly until a condition becomes true, but in such a way that they cancel the others' work out just before they come to test the condition, forcing them both to restart forever. This condition is called:

- A. Deadlock
- B. Race
- C. Livelock.
- D. Virtual race.
- E. Blocking race.

Explain: (use allocated space only)

---

---

---



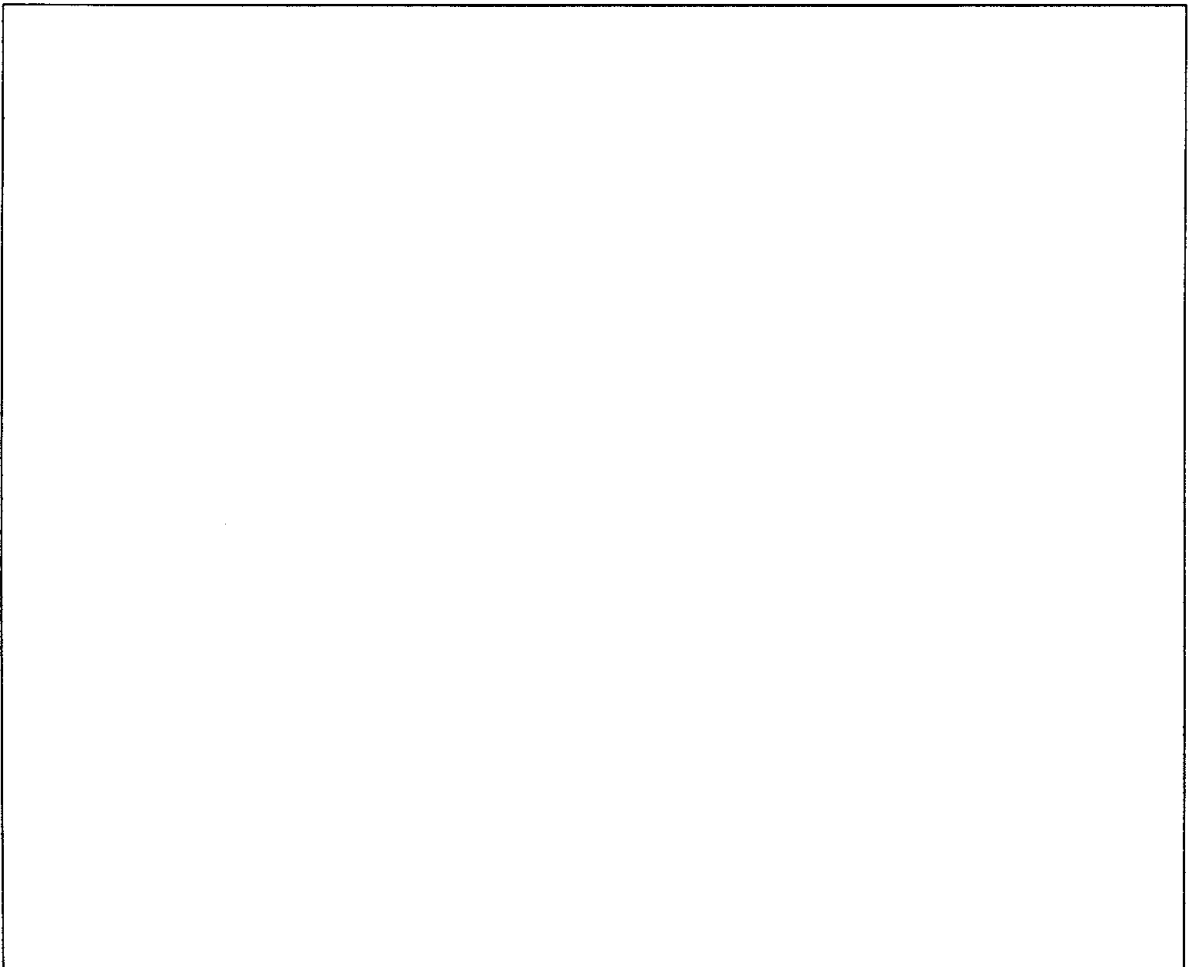
## 2. Class Diagram

From the description above, draw a class diagram which contains the following classes (as well as other classes):

- Spreadsheet
- Row
- Column
- Cell
- Formula

You can assume that the *Formula* class has a method called *calculate* which returns the value of a cell.

**Important:** You must make use of the *Observer* design pattern and indicate clearly where and how you used it.





### 3. Sequence Diagram

Consider the following:

- Cell *A* has a value of 'B+C'.
- Cell *B* has a value of 5.
- Cell *C* has a value of 5.

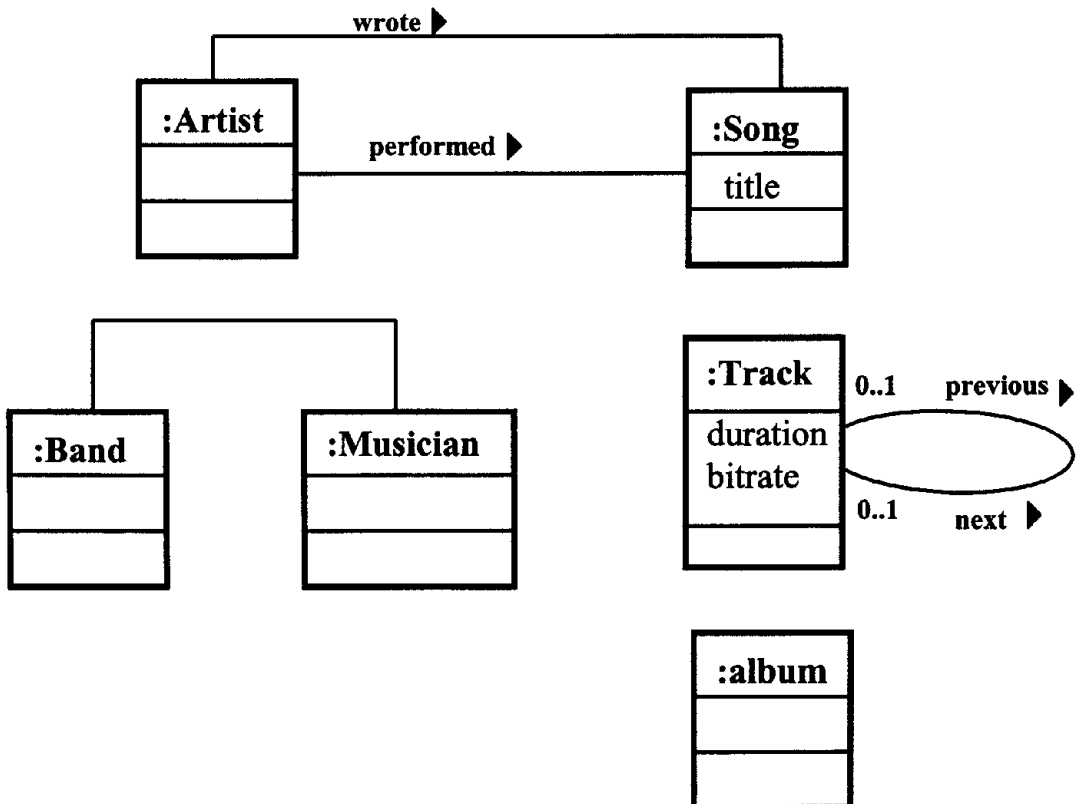
Using your design from part 2, draw a sequence diagram showing what happens when the user changes the value of cell *B* to 10. Indicate clearly on your diagram where the *Observer* pattern is being used.

## Question 2 –UML

(10 marks)

Draw a UML Class Diagram representing the following elements from the problem domain for digital music players: An artist is either a band or a musician, where a band consists of two or more musicians. Each song has an artist who wrote it, and an artist who performed it, and a title. Assume a “song” means a *recording* of a piece of music, so that if a piece of music is recorded more than once (say, by different artists), we treat them as different songs. Therefore, each song is performed by exactly one artist, and written by exactly one artist. An album is composed of a number of tracks, each of which contains exactly one song. A song can be used in any number of tracks, because it could appear on more than one album (or even more than once on the same album!). A track has bit-rate and duration. Because the order of the tracks on an album is important, the system will need to know, for any given track, what the next track is, and what the previous track is (if there is one).

Fill in the details of the partial class diagram below (add associations and multiplicity). Follow UML notations.



### Question 3: Black-box Testing

(10 marks.)

We have a new marking system that takes a .in file as an argument and produces a sorted .out file as specified below.

Input File: The input file format is as follows. Each line will contain the data for a single student. The input will contain the following fields. A semicolon separates each field.

Input fields:

1. Student number (9 digit number)
2. Month of Birth (As a 3 character case insensitive string e.g. Jan, FEB, mar,... Dec)
3. Date of Birth (1-31, but must be valid for the month)
4. First name (String of letters without numbers)
5. First letter of surname (Letter)
6. Number of missed lectures (0-26)
7. Slept during one or more lectures (true/false, written as T or F)

Example:

- Student number: 199226129
- Month of Birth: May
- Date of Birth: 26
- First name: Kevin
- First letter of surname: C
- Number of missed lectures: 0
- Never slept during lectures.

**Input line:** 199226129;May;26;Kevin;C;0;F

If any input row is invalid, the program should print a warning message and continue with the next record. If the program encounters a more serious problem (e.g., unable to open input file), it will print an error message and exit gracefully.

Output File: The output file format is as follows. Each line will contain the data for a single student. The output will contain the following fields. Each field is separated by a semicolon.

Output fields:

1. Student number (9 digit number)
2. Assignment 1 mark: (0-5)
3. Assignment 2 mark: (0-25)
4. Assignment 3 mark: (0-10)
5. Exam mark: (0-60)
6. Final grade: [F, D, C, B-, B, B+, A-, A]
7. Comment: (String of arbitrary length)

Example:

. Student number: 199226129



## Question 4 – Use Case Modeling

(12 marks total. The first four questions are worth 1 mark each and the last four questions are worth 2 marks each.)

Consider the following scenario of a company that offers a web-based service for delivering flowers and gifts. This scenario is called “Ordering flowers delivery”.

The events are:

1. The user selects “flowers” from the menu of possible gifts.
  2. The system presents to the user a variety of price ranges.
  3. The user selects the required price range.
  4. The system presents several pictures of flower arrangements in the selected price range.
  5. The user selects the arrangement it desires.
  6. The system presents the option to send a card as well.
  7. If the user selects sending a card, the system guides the user following the use case “edit card”.
  8. The system offers the user the option of express delivery for extra charge.
  9. The user selects its choice.
  10. The system offers the user to receive a confirmation after the arrangement is received.
  11. The user chooses if a confirmation is needed.
  12. The system presents the details of the order and asks for a confirmation.
  13. The user confirms the order.
  14. The system passed the order to payment and execution.
- 
1. What is the entry condition for the use-case?
    - A. The user is offered several possible gifts.
    - B. The user is presented with a range of prices.
    - C. The user is presented with several flower arrangements.
    - D. The user is presented with the opportunity to add a card.
    - E. None of the above.
  
  2. Which actors are associated with that use-case?
    - A. One actor: the user.
    - B. Two actors: the user and the recipient of the flower arrangement.
    - C. Three actors: the user, the recipient and the system.
    - D. Three actors: the user, the flower arrangement and the system.
    - E. Four actors: the user, the flower arrangement, the recipient and the system.

3. What is the relationship between the use-case “Ordering flowers delivery” and the use-case “edit card” mentioned in step 7?
- A. An alternative.
  - B. An exception.
  - C. Includes.
  - D. Extends.
  - E. Edit card is not a separate use case, but rather another scenario of “Ordering flowers delivery”.
4. Which of the following scenarios would be an exception of the scenario above?
- A. A scenario that is the same except that in step 7 the user chooses not to send a card.
  - B. A scenario that is the same, except that in step 9 the user chooses not to have an express delivery.
  - C. A scenario which is the same except that in step 11 the user does not want a confirmation.
  - D. A scenario where in step 13 the user chooses not to confirm her reservation.
  - E. None of the above.
5. One of the engineers in the team claims that there is no need for a separate use-case per type of gift and that “flower arrangement” can be replaced by a “gift”. What do you think about this claim?
- A. He is right.
  - B. This is possible, but only if any type of gift will have a separate sequence diagram.
  - C. This is possible, but only if the class “gift” will have a sequence diagram that determines the type of gift and the next step of the use case given the gift.
  - D. This is impossible since this would influence the number of actors.
  - E. This is impossible since the complexity of the tests would be too great.
6. During the system testing the images of the arrangements are not available. What is the impact of this fact on the testing?
- A. None.
  - B. Tests cannot be done.
  - C. Steps 4-5 should be omitted from the use case and the system should be tested without them.
  - D. A short textual description can replace the images in describing the arrangements.
  - E. Same as D. But when the images will be available, regression tests will be needed.

7. Which of the following test-cases are needed?

- A. Ordering a flower arrangement without a card.
- B. Ordering a flower arrangement using express delivery.
- C. Ordering a flower arrangement without a confirmation.
- D. All three A-C.
- E. None of A-C is needed.

8. Suppose the user does not select her choice in step 9. Then

- A. The scenario terminates with a failure.
- B. The scenario continues, but the delivery will not be executed that day.
- C. The scenario continues, but there will be a conflict when the system will have to execute the delivery.
- D. The scenario continues, but there might be uncertainty concerning the payment.
- E. The system will wait indefinitely until the user selects her choice.

### Question 5 - Design patterns

(8 marks total)

During the development of a medical imaging application, there is a need to consider medical images from different sources (CT scan, X-Ray, MRI, etc). You identify two problems in the system. The first is that medical images can often be huge and that creating, storing and moving them is extremely costly. The second is that during the development it is not yet clear how and where the images will be located and that there might be different variant of images that will be treated differently.

What design patterns will you use for the images (a combination of two patterns is ideal here)? Justify your choice.

---

---

---

---

---

Draw the UML class diagram:

