



CMPS 282 Final Exam

January 28, 06

2hour exam

Student Name: _____ Student ID: _____

Signature: _____ Section: _____ | _____

There are **eleven** pages, including this one. The test is out of 100 marks, and the value of each question is provided. Please use this information to manage your time effectively.

Question 1: _____/10
Question 2: _____/15
Question 3: _____/10
Question 4: _____/25
Question 5: _____/20
Question 6: _____/25
Total: _____/100

Question 1: True or False [10 marks]

Answer by placing either a **T** or **F** for True and False, respectively. Incorrect answers incur a penalty at a *floor* (2:1) ratio.

A) The following table represents an example of functional testing ___

Use Case	Function Being Tested	Initial System State	Input	Expected Output
Session	System reads a customer's ATM card	System is on and not servicing a customer	Insert a readable card	Card is accepted; System asks for entry of PIN

B) Scenario-based testing, does not depend on use-cases. ___

C) Intra-method testing is similar to intra-procedural testing. ___

D) Component-level design does not consider the component's interface characteristics. ___

E) Cohesion is a qualitative measure of the degree to which components are independent from one another. ___

F) Dataflow testing is a good way to test the Graphical User Interface of an application. ___

G) Airline reservation systems; library catalog systems; and hotel booking systems are an example of Data Centered Architecture. ___

H) Good GUI design/implementation should not contribute to the overall reduction of a user's memory load. ___

I) In moderately complex systems all infrastructure requirements are not initially visible in the problem or business domain. ___

J) The intent of the analysis model is to provide a description of the required information, functional, and behavioral domains for a computer-based system. ___

Question 2 Analysis & Modeling [15 marks]

A) It is well established that use-cases play an important part of the requirements engineering process. Describe their weaknesses. **[3 marks]**

B) What are the steps needed to build an object-behavior model? **[4 marks]**

C) List the elements of the structured analysis model and explain the role of each element. **[3 marks]**

D) An ATM machine can be started or shut down by the bank. The ATM machine will service one client at a time. To interact with the ATM machine, the client initiates a session by inserting an ATM card and enters a personal identification number (PIN). The data will then be sent to the bank for validation as part of each transaction. Once authenticated, the client is able to withdraw, deposit to any account linked to the card and enter the amount of the deposit into the ATM. A client is also able to make money transfer between any two accounts linked to the card, and inquire about the balance. A client is always able to abort a transaction in progress by pressing the **Cancel** key. Every transaction, including the authentication is verified by the bank. Create a use case analysis model of the above system. **[5 marks]**

Question 3 Design [10 marks]

A) How does the object-oriented view of component-level design differ from the traditional view? [5 marks]

B) Describe the differences between a design pattern and a framework. Give examples to illustrate your answer. [5 marks]

Question 4 Design Patterns [25 marks]

A) Which pattern is more appropriate to use, when objects (from various classes in a hierarchy) need to be generated at runtime. Give an OO example. Code is required. **[8 marks]**

B) Give an example of the use of singleton pattern. OO Code is required. **[8 marks]**

C) Give a scenario where the use of the observer pattern is desired. Give an OO example. Code is required. **[9 marks]**.

Question 5 GUI [20 marks]

A) Describe the main differences between mental and conceptual models. [5 marks]

B) On Page 8/10 are some of the windows and menus from a program for a computerized appointment book. A description of the functionality of this program is given below. Very briefly, and in point form, identify the problems in this interface. For each problem, you should indicate which of the design principles below is violated. Do not recommend ways to improve the interface — just identify the problems. [15 marks]

DESIGN PRINCIPLE

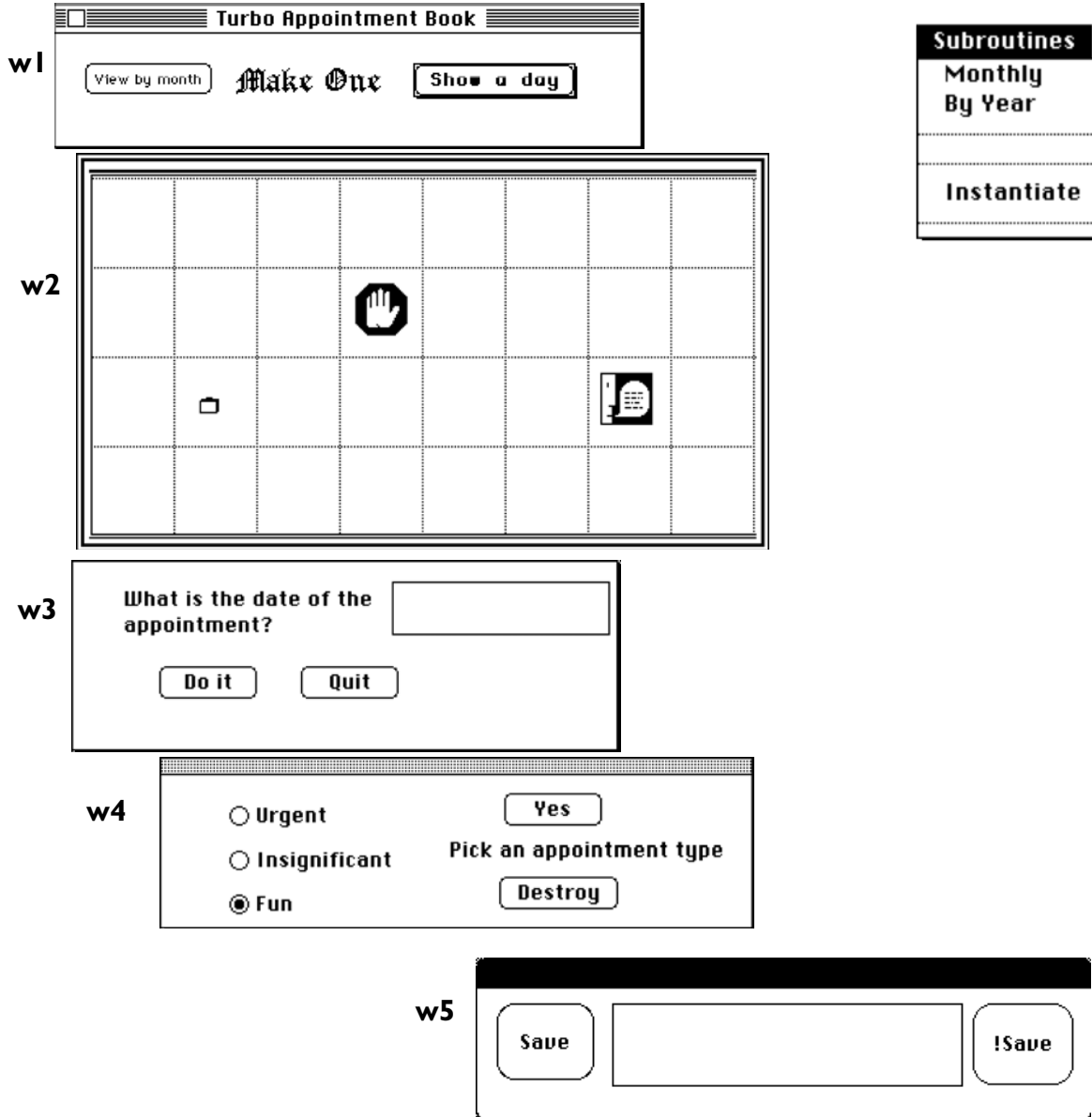
- i) Use both knowledge in the world and in the head.
- ii) Simplify the structure of tasks.
- iii) Make things visible.
- iv) Get the mappings right.
- v) Exploit the power of constraints.
- vi) Design for error.
- vii) Standardise.

INTERFACE DESCRIPTION

This program provides the following functionality:

- Can record appointments for any time/day/month/year.
- Allows viewing of appointments by day, month or year.
- Provides for different priorities of appointments.

When run, the program displays **Window 1**. There are standard menus for **File** and **Edit** (not shown), and a third menu, called **Subroutines**






Creating Appointments

To add a new appointment, press **Make One**, in **Window 1**, or select **Instantiate** from the **Subroutines** menu. This brings up **Window 3**. Here the time and date of the appointment should be entered in the following format: *month-time:day/year*. If the input is not in the format, pressing either button causes the computer to beep, and do nothing. If it is in a valid format, pressing **Quit** closes the window and cancels the addition of the new appointment, and pressing **Do it** closes **Window 3** and opens **Window 4**. In **Window 4** the priority of the appointment can be recorded as **Urgent**, **Insignificant** or **Fun**. Pressing **Destroy** cancels the addition of the new appointment. Pressing **Yes** closes **Window 4** and opens **Window 5**. In **Window 5** the user enters information concerning the appointment in the text field. Pressing **Save** saves the

appointment and closes **Window 5**. Pressing **!Save** saves the appointment and discards any other appointments which had been defined on the same day.

Viewing Appointments

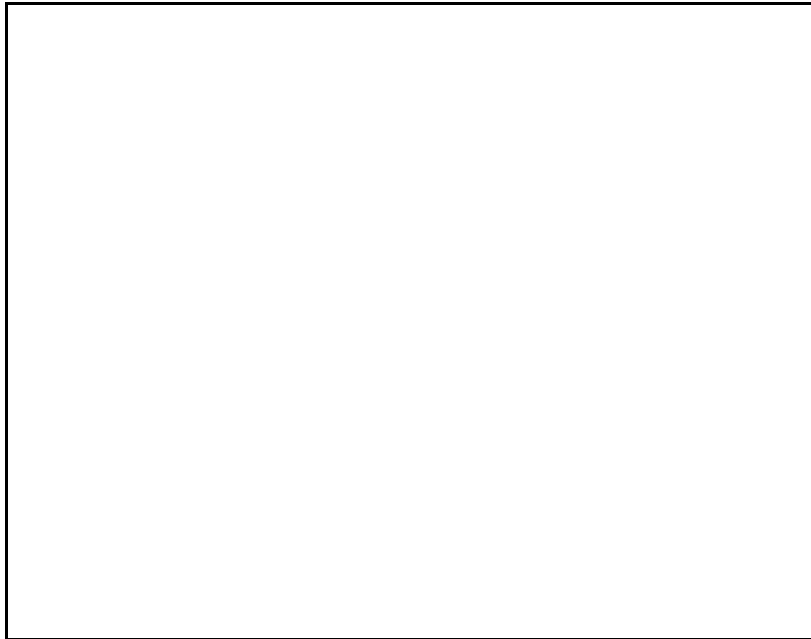
To view appointments by day, press the button **Show a day** in **Window 1**. To view by month, press the button **View by month**, or select the menu item **Monthly**. To view by year, select the menu item **By Year**. To keep this question from getting too long, only viewing by month is described here, and you do not have to design view by day and by year for your interface. View by month is shown in **Window 2**. To select which month to view, type in the number of the month immediately after pressing View by month, or selecting the **Monthly** menu item. Each box in the grid represents a day. The icons indicate days with appointments, with the type of icon indicating the priority as follows:  = Fun,  = Insignificant, and  = Urgent. Double clicking on a day brings up the view by day window for that day.

Question 6 Testing [25 marks]

Given a control flow graph (CFG) $G(V,E)$, where V is the number of vertices, and E the number of edges.

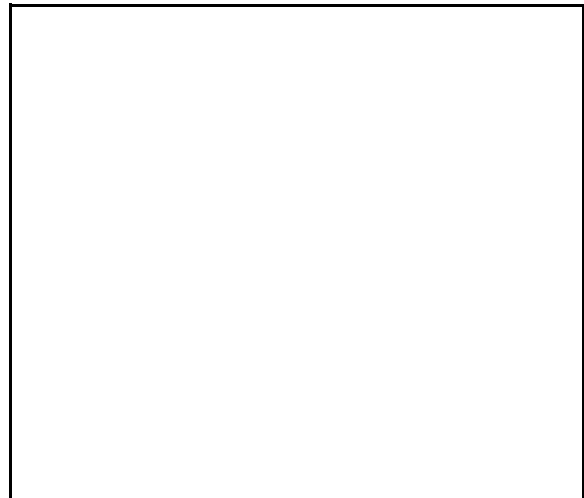
A) Draw the CFG for the following code: [3 marks]

```
Void main() {  
  int a,b;  
  a=5;  
  b=a/2;  
  if(b < 3) {  
    a--;  
    printf("%d", a);  
  }  
  else {  
    b--  
    printf("%d", b);  
  }  
}
```



B) Extract the all-dus with regards to variables a and b . Also explain dead code. [4 marks]

C) Give a C example that shows, and prove that all-node testing is weaker than all branches. You should draw the CFG [4 marks].



D) What does the cyclomatic complexity of a CFG indicate. **[2 marks]**

E) A lower cyclomatic complexity is an indication of what? elaborate. **[1 marks]**

F) What are the challenges of OO testing. Give an example of each. **[5 marks]**

G) What is the difference between bottom-up and top-down testing. Explain **[2 marks]**

H) What is inter-class testing? Explain. **[4 marks]**
