The University of Texas at Dallas
Computer Science Program

Test 2 April 27, 2000

Conditions: Closed book Duration: 70 minutes

Write legibly; unreadable answers are not answers!

Each answer should be concise and precise! Indicate any assumptions

Name: ______________________________

{Please underline last name}

Student Number: ______________________________

1. __________ /20

2. __________ /30

3. __________ /20

4. __________ /10

5. __________ /20

Total __________ /100

1. [20 marks]

For each of the following ten statements, indicate whether it is true (mark T) or false (mark F). (No penalty for a wrong answer)

__T__ This is the second test for CS6362.

____ 1. In order to enable users of a distributed system to use plug-and-play, a clear definition of its components and constraints is both necessary and sufficient.

____ 2. In order to improve interoperability among system architectural components, it is desirable to have a set of proprietary APIs for programming languages.

____ 3. As the high-level solution to the given problem, a novel role of software architecture lies in change management: change in requirements does not necessarily imply change in the architectural definition.

____ 4. In the context of DAD, communication based on MOM is considered more conceptual than communication based on sockets but less formal than communication based on RPC.

____ 5. A software architecture to be implemented using the event feature of Java can no longer have any objects as its components.

____ 6. MILs are expressively as powerful as ADTs, especially in consideration of Z.

____ 7. Especially when the environment frequently changes, a pipe-and-filter architecture should be chosen for an effective feedback mechanism.

____ 8. In a file-based client/server architecture, the primary interface is for manipulating chunks of primitive data types (e.g., integer).

____ 9. A key strength of the web client/server architecture is that it allows the processing load to be distributed well between the client and the server.

____ 10. The degree of formality in a software architectural design is the primary determinant of the quality of the design, as no contextual analysis is possible without formality.

2. [**30 marks**]

Circle the best answer to each of the following questions.

1. Which is the least relevant to execution of RPCs?

1. message queue blocking
2. result marshalling
3. server stub
4. parameter unmarshalling
5. transport entity

2. Which is the most relevant to a client/server architecture with database servers?

1. events
2. TP monitor
3. URL
4. object bus
5. all of the above

3. In a client/server groupware architecture, the central concern is

1. inheritance
2. email transmission
3. multimedia
4. events
5. none of the above

4. Which is the least relevant to a transaction?

1. data persistence
2. process persistence
3. indivisible unit of work
4. non-interruptible
5. consistency

5. Which is the most relevant to object-oriented component definition?

1. socket connection
2. interpreter
3. marshalling
4. encapsulation
5. port id

Circle the best answer to each of the following questions.

6. Which would be a major emphasis in an object-oriented distributed processing architecture?

1. vertical integration
2. proprietary network architecture
3. compound documents
4. fat server
5. service and interface specification

7. Which is the least relevant to a blackboard architecture?

1. knowledge source
2. indirect visibility
3. control
4. signal processing
5. problem-solving state data

8. Which is the most relevant to MOMs?

1. middleware-based solutions
2. groupware specification
3. device configuration
4. fixed sequence of transmissions
5. lock-step communication

9. Which is the least relevant to the collaborative computing paradigm?

1. clients and servers
2. knowledge sharing
3. centralized system
4. achieving common goals
5. smart agents

10. Which is the most relevant to client-server architecture according to the working definition in class?

1. a unique server for each client request
2. homogeneity of client applications
3. visibility of each server
4. interface specification using equational algebra
5. implementation-independent specification

3. [20 marks]

3.1. Consider a software architecture for an event-based agent intended to help used-car buyers and sellers. *Precisely and concisely* show an architectural diagram, using the convention as discussed in class, and describe what events mean.

3.2. *Precisely and concisely* describe how the architecture in **3.1** can be implemented using JavaBeans, as discussed in class. Give a diagram and an explanation.

3.3. Using a diagram, precisely and concinsely explain the similarity between a "Christian marriage" and the two-phase protocol for a client/server architecture with TP monitors.

3.4. Precisely and concisely describe two dominant techniques for recovery from a transaction failure (at a server). Describe also how they can be implemented.

4. [10 marks]

4.1. Using an example of an employee database, precisely and concisely describe how to enforce the "all-or-nothing" proposition when using a client/server architecture with database processing.

4.2. Using an example, precisely and concisely describe how an object management system is different from a database management system.

5. [20 marks]

Consider the following definition:

```
Voltage == Z X {Volt}           /* "X" refers to a cross product */
Time == N X {Second}

Waveform == Time -|-> Voltage    /* "-|->" refers to a partial function */

Segment == {w: Waveform | dom w subsetOf ContigPeriod},
            where ContigPeriod == {st: P1 Time| (exists t1, t2: Time .
                                       st = {t: Time| t1<t<t2})}
Coord == R X {Metre}
Point == Coord X Coord
Trace == Time -|-> Point


___ Scale ____________________________________________________________
| segment: Segment
| HScale: Time
| VScale: Voltage
| scaled: Trace
----------------------------------------------------------------------
| scaled =   (lambda t: dom seqment .
|                     ( (t - min(dom segment)) / HScale,
|                       (segment(t) - min(ran segment))/VScale) )
|                       /* "dom" refers to domain, "ran" refers to range */
----------------------------------------------------------------------


___ Translate _______________________________________________
| scaled: Trace
| HOffset, VOffset: Coord
| moved: Trace
-------------------------------------------------------------
| moved = (lambda t: dom scaled . (second(scaled(t)) - HOffset,
|                                   first(scaled(t)) - VOffset))
--------------------------------------------------------------


__ Clip ______________________________________________________
| moved: Trace
| clipped: Trace
--------------------------------------------------------------
| let screen == {(x, y): Point| 1 < x < HMax and 2 < y < VMax} .
|                 clipped = moved |> screen   /* "|>" is the range restrictor
--------------------------------------------------------------
```

In the following, measurement units will be omitted. Also note that "|–" denotes mapping at the instance level. Assume

```
segment = {5  |-  7, 6  |-  9, 7  |-  7, 8  |-  2, 9  |-  5, 10  |-  4}
HScale = VScale = 1
HOffset = 1
VOffset = 2
HMax = 8
VMax = 10
```

5.1. Compute the value of "scaled".

5.2. Compute the value of "moved" corresponding to the above.

5.3 Compute the value of "clipped".

5.4 Define a schema "DisplayTraces" which can be used to display both "moved" and "clipped" traces on the display at the same time.