14.01.20

09.30 - 11.30am

CMPU 4021 Distributed Systems

KE G43 Kevin Street

Programme Code: DT211C, DT228, DT282
Module Code: CMPU 4021
CRN: 32475, 22417, 25775

TECHNOLOGICAL UNIVERSITY DUBLIN

KEVIN STREET CAMPUS

BSc. (Honours) Degree in Computer Science (Infrastructure)

BSc. (Honours) Degree in Computer Science

BSc. (Honours) Degree in Computer Science (International)

Year 4

SEMESTER 1 EXAMINATIONS 2019/20

Distributed Systems

Dr. Edina Hatunic-Webster
Dr. Deirdre Lillis
Dr. David Malone – DT211C
Mr. Patrick Clarke – DT228/DT282

Two Hours

Attempt **3 questions**
All questions carry **equal** marks
One complimentary mark is available

1. **(a)** IP provides a *multicast* facility. Explain what this means, then discuss the scenarios where multicast is used.

(8 marks)

(b) A server program written in one language (for example C) provides the implementation of an object that is intended to be accessed by clients that may be written in a different language (for example Java). The client and server computers may have different hardware, but all of them are attached to an internet.

Discuss the problems due to each of the five aspects of *heterogeneity* that need to be solved to make it possible for a client object to invoke a method on the server object.

(12 marks)

(c) Provide Java code to multicast a message from a client to a group.

(13 marks)

2. **(a)** Explain the characteristics of a *peer-to-peer* systems and give examples of *peer-to-peer middleware* platforms.

(8 marks)

(b) Discuss the three alternative approaches to *external data representation* and marshalling.

(12 marks)

(c) Provide sample Java code that show how to create a TCP server that uses *Java serialisation* to receive a Java object of type *Person* where the Person class has *name*, *address* and *personID* attributes.

(13 marks)

3. (a) A distributed *banking* system has *account* objects available to connecting clients. These remote objects allow a client to:

- Get the name of the account owner
- Get the balance
- Make deposits
- Make withdrawals

Define the *interface* to the account service in *Java Remote Method Invocation (RMI)*.

(8 marks)

(b) With the help of a diagram explain *dynamic class loading* in Java RMI. In your answer outline the security steps required to allow a Java RMI application to securely use dynamic class loading.

(12 marks)

(c) Managing a *distributed transaction* involves many transaction managers and is made difficult because the state of one system can change suddenly, unknown to the other participants in the transaction.

Using examples and diagrams *show* how the *two phase commit* protocol addresses the problem outlined here.

(13 marks)

4. (a) Explain the purpose of UDDI. What are the *four* different data structures that support UDDI?

(8 marks)

(b) Suggest a design for *a notification mailbox* service that is intended to store notifications on behalf of multiple subscribers, allowing subscribers to specify when they require notifications to be delivered. Explain how subscribers that are not always active can make use of the service you designed.
How will the service deal with subscribers that crash while they have delivery turned on?

(12 marks)

(c) With the help of a diagram describe Web services infrastructure and components. Provide an analysis of its strengths and weaknesses when compared with competing approaches.

(13 marks)