

UNIVERSITI SAINS MALAYSIA

Peperiksaan Semester Pertama
Sidang Akademik 2004/2005

Oktober 2004

CCS521 – Konsep dan Reka Bentuk Sistem Teragih Maju

Masa : 2 jam

ARAHAN KEPADA CALON:

- Sila pastikan bahawa kertas peperiksaan ini mengandungi **EMPAT** soalan di dalam **EMPAT** muka surat yang bercetak sebelum anda memulakan peperiksaan ini.
 - Jawab **SEMUA** soalan.
 - Anda boleh memilih untuk menjawab semua soalan dalam Bahasa Malaysia atau Bahasa Inggeris.
-

1. (a) Algoritma teragih boleh dibahagikan kepada dua jenis; satu jenis yang memerlukan satu proses khas sebagai koordinator manakala yang satu jenis lagi tidak mempunyai koordinator khas.
 - (i) Beri satu contoh algoritma untuk setiap jenis. (2/25)
 - (ii) Bincangkan kekuatan dan kelemahan utama kedua-dua jenis algoritma ini. (4/25)
 - (b) Kenapa kadangkala menjadi sukar untuk melindungi kejadian (occurrence) dan pemulihan kesilapan di dalam sesuatu sistem teragih? (4/25)
 - (c) (i) Sebut **dua (2)** keperluan utama sesuatu perkhidmatan nama. (2/25)
 - (ii) Apakah masalah utama dengan navigasi multicast untuk perkhidmatan nama? (2/25)
 - (iii) Cadang **dua (2)** cara untuk mengatasi masalah di 1(c)(ii). (6/25)
 - (iv) Cadang satu kaedah untuk mengatasi masalah maklumat basi yang disediakan oleh resolver nama tempatan yang menggunakan teknik cache. (5/25)
2. (a) Pertimbangkan suatu kawasan kritikal di dalam suatu gelung. Andaikan terdapat data yang dikongsi di dalam kawasan kritikal tersebut direplikasikan pada nod-nod lain.
 - (i) Bezakan antara "eager release consistency model" dan "lazy release consistency model" berdasarkan situasi di atas. (6/25)
 - (ii) Nilai kedua-dua model di 2(a)(i) dari segi kegunaan bandwidth dan bilangan mesej yang dihasilkan. (9/25)

- (b) Pertimbangkan pelaksanaan dua proses berikut. Andaikan nilai awal semua pemboleh ubah adalah sifar.

P1: R(x)1; R(x)2; W(y)1

P2: W(x)1; R(y)1; W(x)2

- (i) Adakah ingatan untuk pelaksanaan di atas konsisten secara jujukan? Terangkan jawapan anda.

(5/25)

- (ii) Adakah ingatan untuk pelaksanaan di atas koheren? Terangkan jawapan anda.

(5/25)

3. Antara muka *Election* mempunyai dua fungsi terpencil:

vote: terdapat dua parameter iaitu nama (*string*) dan nombor pengundi (*integer* yang akan memastikan hanya satu undi sahaja untuk setiap pengundi), yang diberi oleh pelanggan.

result: terdapat dua parameter iaitu nama calon dan jumlah undi, yang diberi oleh pelayan kepada pelanggan.

- (a) Yang mana parameter-parameter daripada dua fungsi tersebut yang merupakan parameter input dan parameter output?

(4/25)

- (b) Takrifkan antara muka *Election* dalam CORBA IDL dan JAVA RMI. Nyatakan bagaimana fungsi CORBA IDL dan fungsi JAVA RMI berbeza dari segi *input* argumen dan *output* argumen.

(10/25)

- (c) Servis *Election* mesti memastikan hanya satu undi direkodkan apabila pengguna fikir mereka telah mengundi. Bincangkan kesan semantik mungkin dipanggil untuk servis *Election*. Adakah satu panggilan semantik 'sekurangnyanya' diterima untuk servis *Election* atau adakah anda cadangkan satu panggilan semantik 'selebihnya'?

(5/25)

- (d) Rangkakan satu implementasi untuk servis *Election* yang memastikan rekod konsisten apabila dicapai pada masa yang sama oleh beberapa pelanggan.

(6/25)

4. (a) Satu pelanggan membuat panggilan prosedur terpencil kepada pelayan. Pelanggan mengambil 10 milisaat untuk proses parameter setiap permintaan, dan pelayan mengambil 20 milisaat untuk setiap permintaan. Masa pemprosesan sistem pengoperasi tempatan untuk setiap operasi hantar dan terima ialah 1 milisaat, dan masa rangkaian untuk menghantar setiap mesej permintaan atau balasan ialah 5 milisaat. "Marshalling" atau "unmarshalling" setiap mesej mengambil 1 milisaat.

Kira masa yang diambil oleh pelanggan untuk menghasil dan mengembalikan dua permintaan:

- (i) Jika beban tunggal, dan (5/25)
- (ii) Jika dua beban yang boleh membuat lebih daripada satu permintaan pada masa sama dalam satu pemproses. (5/25)
- (b) Bandingkan penjadualan static dengan penjadualan dinamik. Beri satu contoh kepada setiap penjadualan? (5/25)
- (c) Kemukakan masalah-masalah penyelidikan sekarang dalam bidang grid pengetahuan. (5/25)
- (d) Bandingkan proses pencarian dalam Napster dengan GNUTELLA. (5/25)

UNIVERSITI SAINS MALAYSIA

First Semester Examination
Academic Session 2004/2005

October 2004

CCS521 – Advanced Distributed Systems Concepts and Design

Duration : 2 hours

INSTRUCTION TO CANDIDATE:

- Please ensure that this examination paper contains **FOUR** questions in **FOUR** printed pages before you start the examination.
 - Answer **ALL** questions.
 - You can choose to answer either in Bahasa Malaysia or English.
-

ENGLISH VERSION OF THE QUESTION PAPER

1. (a) Distributed algorithms can be classified into two; one requiring the use of a coordinating process while the other has a non-fixed coordinator.
 - (i) State an example algorithm for each class. (2/25)
 - (ii) For each, discuss the main advantage and disadvantage of the algorithm. (4/25)
 - (b) Why is it sometimes difficult to hide the occurrence and recovery from failures in a distributed system? (4/25)
 - (c) (i) State **two (2)** necessary requirements of a name service. (2/25)
 - (ii) What is the main problem with multicast navigation in name services? (2/25)
 - (iii) Suggest **two (2)** ways to overcome the problem in 1(c)(ii). (6/25)
 - (iv) Propose a way to overcome the problem of stale information provided by local name resolvers that use caching. (5/25)
2. (a) Consider a critical region which is located inside a loop. Assume that within the critical region, there are shared data which are replicated at various nodes.
 - (i) Differentiate between eager release consistency model and lazy release consistency model using the above scenario as illustration. (6/25)
 - (ii) Evaluate both models in 2(a)(i) in terms of bandwidth utilization and number of messages generated. (9/25)

- (b) Consider the following execution of two processes. Assume that initially, all variables are set to zero.

P1: R(x)1; R(x)2; W(y)1

P2: W(x)1; R(y)1; W(x)2

- (i) Is the memory underlying the above execution sequentially consistent? Explain your answer.

(5/25)

- (ii) Is the memory underlying the above execution coherent? Explain your answer.

(5/25)

3. The *Election* interface provides two remote methods:

vote: with two parameters through which the client supplies the name of a candidate (a string) and the voter's number (an integer used to ensure each user votes once only). The voter's numbers are allocated sparsely from the range of integers to make them hard to guess.

result: with two parameters through which the server supplies the client with the name of a candidate and the number of votes for that candidate.

- (a) Which of the parameters of these two procedures are input parameters and which are output parameters?

(4/25)

- (b) Define the interface to the *Election* service in CORBA IDL and JAVA RMI. Compare the methods in the two languages for specifying *input* and *output* arguments.

(10/25)

- (c) The *Election* service must ensure that a vote is recorded whenever any users think they have cast a vote. Discuss the effect of 'maybe' call semantics on the *Election* service. Would 'at-least-once' call semantics be acceptable for the *Election* service or would you recommend 'at-most-once' call semantics?

(5/25)

- (d) Outline an implementation for the *Election* service that ensures that its records remain consistent when service is accessed concurrently by multiple clients.

(6/25)

4. (a) A client makes remote procedure calls to server. The client takes 10 milliseconds to compute the arguments for each request, and the server takes 20 milliseconds to process each request. The local operating system processing time for each send or receive operation is 1 milliseconds, and the network time to transmit each request or reply message is 5 milliseconds. Marshalling or unmarshalling takes 1 milliseconds per message.

Calculate the time taken by the client to generate and return from two requests:

- (i) If it is single-threaded, and (5/25)
- (ii) If it has two threads that can make requests concurrently on a single processor. (5/25)
- (b) Compare the static scheduling with the dynamic scheduling. Give one example for each of the scheduling. (5/25)
- (c) Highlight some existing research problems on knowledge grid. (5/25)
- (d) Compare the searching process in Napster and GNUTELLA. (5/25)