ชื่อ_____ รหัสนักศึกษา _____

มหาวิทยาลัยสงขลานครินทร์

คณะวิศวกรรมศาสตร์

ภาควิชาวิศวกรรมคอมพิวเตอร์

สอบกลางภาค: ภาคการศึกษาที่ 2 ปีการศึกษา: 2557

วันที่สอบ: 18 มีนาคม 2558 เวลาสอบ: 13.30 – 16.30

รหัสวิชา: 241-421 ห้องสอบ:

ชื่อวิชา: CLIENT/SERVER DISTRIBUTED SYS

คำสั่ง: อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

อนุญาต: เครื่องเขียนต่าง ๆ

ไม่อนุญาต: เครื่องคิดเลข และ เอกสารใด ๆ

เวลา: 3 ชั่วโมง (180 นาที)

คำแนะนำ:

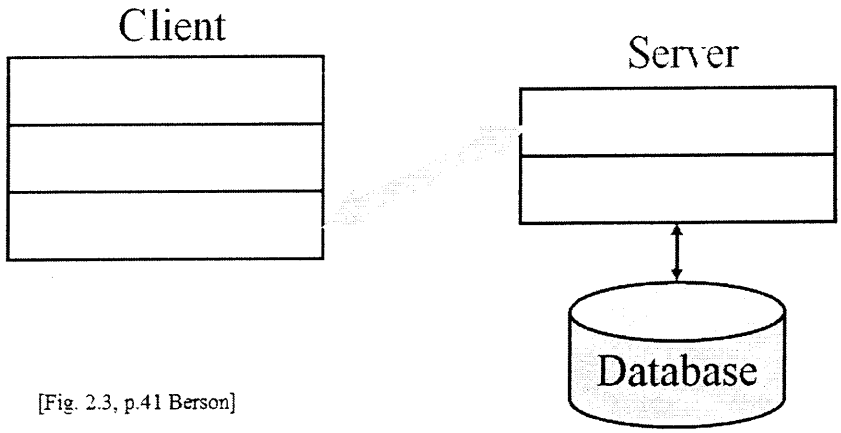
- ข้อสอบมี 12 หน้า (รวมใบปะหน้า) แบ่งเป็น 8 ข้อ คิดเป็นคะแนนเก็บ 20 %
- คำตอบทั้งหมดจะต้องเขียนลงในข้อสอบ
- เขียนชื่อ รหัสนักศึกษา ในทุกหน้าของข้อสอบให้ชัดเจน

ทุจริตในการสอบ โทษขั้นต่ำคือ
ปรับตกในรายวิชาที่ทุจริต และพักการเรียน 1 ภาคการศึกษา

ชื่อ_____ รหัสนักศึกษา _____

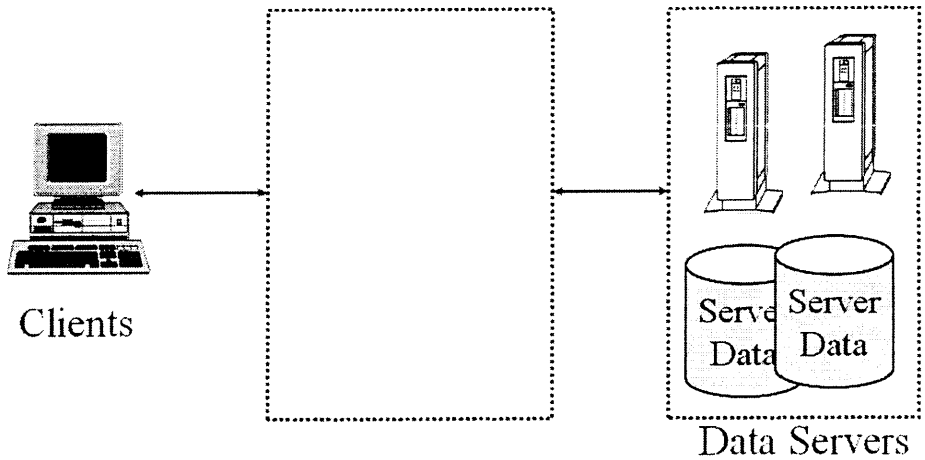
1. The Client Server Model – 4 คะแนน

1.1 จากรูปที่กำหนดให้ จงเติมช่องว่างที่ขาดหายไป พร้อมอธิบายว่าเป็นโมเดลแบบใด



[Fig. 2.3, p.41 Berson]

1.2 จากรูปที่กำหนดให้ จงเติมช่องว่างที่ขาดหายไป พร้อมอธิบายว่าเป็นโมเดลแบบใด



ชื่อ_____ รหัสนักศึกษา _____

1.3 จงวาดรูปแสดง DCE Client/Server Model พร้อมอธิบายหลักการทำงาน

1.4 จงบอกข้อดีข้อเสียของระบบ Client/Server

ชื่อ_____ รหัสนักศึกษา _____

2. Distributed Programming Concepts - 2 คะแนน

จงอธิบาย Algorithmic Distribution ในรูปแบบต่อไปนี้พร้อมวาดภาพประกอบ

2.1 แบบ Several Workers per Sub-task

2.2 แบบ Parallelism Separate Sub-tasks

ชื่อ _____ รหัสนักศึกษา _____

3. Peer-to-Peer Technologies (P2P) - 2 คะแนน

3.1 จงบอกเป้าหมายของระบบแบบ P2P

3.2 จงอธิบายความแตกต่างของ P2P แบบ Reader-Centric และ Publisher-Centric

ชื่อ_____ รหัสนักศึกษา _____

4. Low-level File I/O - 1 คะแนน

จงอธิบายความสำคัญของการเข้าถึงข้อมูลแบบ *low-level*

5. Processes - 1 คะแนน

จากจงอธิบายหลักการสื่อสารระหว่าง processes พร้อมยกตัวอย่างกรณีสื่อสารผ่านไฟล์

ชื่อ _____ รหัสนักศึกษา _____

6. Networking Concepts - 3 คะแนน

6.1 เพื่อระบุตำแหน่งของ Server จำเป็นต้องคำนึงถึงเรื่องใดบ้าง

6.2 จงวาดไดอะแกรมแสดงการรับส่งของโปรโตคอลแบบ TCP พร้อมอธิบาย

6.3. Client จะส่งความต้องการไปยัง Server ได้อย่างไรจงอธิบาย

7. Sockets - 4 คะแนน

7.1 จงอธิบายชนิดของ Server พื้นฐาน 2 ประเภท

ชื่อ _____ รหัสนักศึกษา _____

7.2. จากโปรแกรมที่กำหนดให้ จงเขียน Basic client-server Operation

```
#define PORT 1666

int main()
{ int rdsock, cdsock;
          :
  rdsock = tcp_serv_sock(PORT);
  while(1) {
    cdsock = accept(rdsock, ...);
    if (cdsock < 0)
      fprintf(stderr, "accept failed\n");
    else {
      do_something(cdsock, ...);
      close(cdsock);
    }
  }
  return 0;
}
```

7.3. จากโปรแกรมในข้อ 7.2 หากกำหนดให้ฟังก์ชัน do_something มีการทำงานต่อไปนี้

```
time(&now);
now_str = ctime(&now);
write(cdsock, now_str, strlen(now_str));
```

จงอธิบายว่า Server นี้มีบริการชนิดใด

7.4 จงบอกชื่อโปรแกรมตัวอย่างที่มีในสไลด์เรื่อง Socket มาอย่างน้อย 4 ชื่อ

ชื่อ_____ รหัสนักศึกษา _____

8. จงเติมส่วนของโปรแกรมให้สมบูรณ์ - 3 คะแนน

```
#include <stdio.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h>
#include <ctype.h>   /* for toupper() */

#define PORT _____(1)
#define BUFSIZE 128

int tcp_serv_sock(int port);
int main()
{
  int rdsock, cdsock, client_len;
  struct sockaddr_in client;

  rdsock = _____(2)

  while(1) {
    client_len = sizeof(client);
    cdsock = accept(rdsock, (struct sockaddr *)&client, &client_len);
if (cdsock < 0)
      fprintf(stderr, "accept failed\n");
    else {
      _____(3)
      close(cdsock);
    }
  }
  return 0;
}
void echo_upper(int sd)
{
  char buf[BUFSIZE];
  int n;

  while (((n = read(sd, buf, sizeof(buf)) != 0) && (!_____(4) ))
    if (n < 0)
      fprintf(stderr, "echo read error\n");
    else {
/*    buf[n] = '\0'; printf("n: %d, buf: \"%s\"\n", n, buf); */
      _____(5)
      if (write(sd, buf, n) < 0)
        fprintf(stderr, "echo write error\n");
    }
}

void touppers(char buf[])
{
  int i = 0;

  while (buf[i] != '\n') {
    buf[i] = toupper(buf[i]);
    i++;
  }
}
```

```
int end_input(char buf[], int len)
/* Check if buf[] contains the characters meaning end-of-input for this server.
Return 1 if true. */
{
  if ((len == 2) &&
      (buf[0] == '\015') && (buf[1] == '\n'))
    return 1;

  if ((len == 1) && (buf[0] == '\n'))
    return 1;

  return 0;
}
```