PLEASE HAND IN

PLEASE HAND IN

UNIVERSITY OF TORONTO AT MISSISSAUGA
APRIL EXAMINATIONS 2004

CSC 309H1
Mississauga Campus

Duration — 3 hours

Examination Aids: ***Two double sided*** $8\frac{1}{2} \times 11$ ***aid sheets. A non-programmable calculator.***

Student Number: |_|_|_|_|_|_|_|_|_|_|_|

Last Name: _____

First Name: _____

Lecture Section: L5101
(A. Rosenbloom)

---

*Do **not** turn this page until you have received the signal to start.*
(In the meantime, please fill out the identification section above, and read the instructions below *carefully*.)

This final examination consists of 5 questions on 13 pages (including this one), printed on both sides of the paper. *When you receive the signal to start, please make sure that your copy of the examination is complete.* Answer each question directly on the examination paper, in the space provided.

Be aware that concise, well thought-out answers will be rewarded over long rambling ones. Also, unreadable answers will be given zero (0) so write legibly.

General Hint: We were careful to leave ample space on the examination paper to answer each question, so if you find yourself using much more room than what is available, you're probably missing something. Also, remember that hints are just hints: you are not required to follow them if you can think of a different solution.

Good Luck!

1: _____/10

2: _____/10

3: _____/10

4: _____/14

5: _____/10

TOTAL: _____/54

Question 1. [10 MARKS]

Part (a) [4 MARKS]

LangerCorp.com has been assigned the class C address 198.77.116.0. and has decided to divide their address space into 4 equally sized subnets (s1,s2,s3,s4).

Complete the following table describing the four subnetworks:

```
Subnet s1: 198.77.116.0  -_________________    Subnetmask:___________________

Subnet s2: _____________-_________________    Subnetmask:___________________

Subnet s3: _____________-_________________    Subnetmask:___________________

Subnet s4: _____________-_________________    Subnetmask:___________________
```

Hints:

- s1 should consist of a contiguous block of IP addresses, similarly for s2, s3 and s4.
- **Network:** can be computed by **anding** the subnetmask and an IP address. For example, both of the following IP addresses are on the same class C network 210.5.7.0 (with subnetmask 255.255.255.0).

```
  210.  5.  7. 11          210.  5.  7.108
& 255.255.255.  0        & 255.255.255.  0
-----------------        -----------------
  210.  5.  7.  0    =     210.  5.  7.  0
```

Part (b) [6 MARKS]

Complete the following IP Routing table for host h1 (SEE NEXT PAGE)

```
Kernel IP routing table
Destination          Gateway              Genmask              Iface
```

```
210.5.7.0            0.0.0.0              255.255.255.0        e0
127.0.0.0            0.0.0.0              255.0.0.0            lo
```

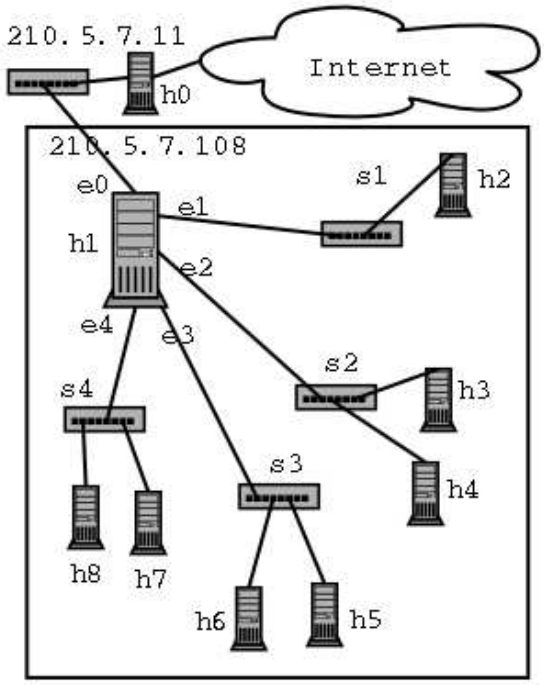


Figure 1: LangerCorps Network

- LangerCorp.com's main gateway (denoted **h1** in the diagram) is given IP address 210.5.7.108 on their ISP's 210.5.7.0 network. (ISP stands for Internet Service Provider)
- The ISP's main gateway is **h0** and has IP address 210.5.7.11 on the 210.5.7.0 network.
- LangerCorp.com has given IP 210.5.7.108 to interface **e0** on host **h1**.
- **Destination:** is the destination network
- **Genmask:** is the subnet mask
- **Iface:** is the interface (also known as ethernet card) to use
- The IP Routing table we discussed in class is shown below

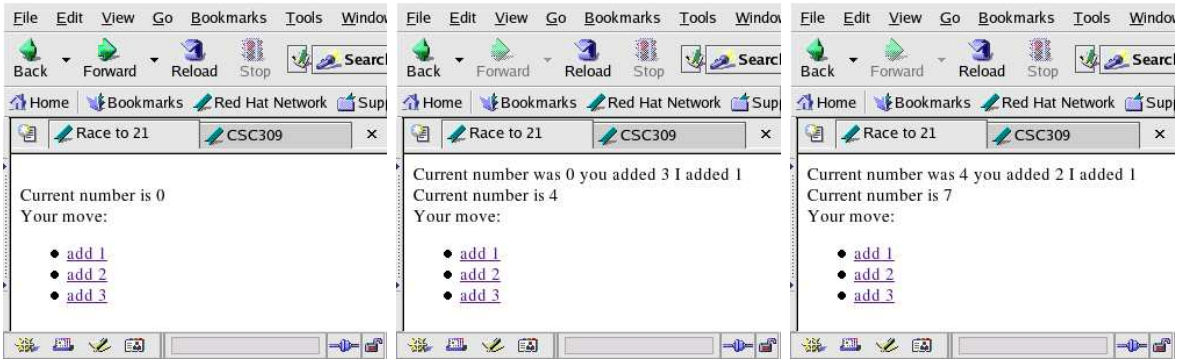
```
Destination     Gateway         Genmask         Iface
192.168.1.0     0.0.0.0         255.255.255.0   eth1
142.150.8.0     0.0.0.0         255.255.252.0   eth0
127.0.0.0       0.0.0.0         255.0.0.0       lo
0.0.0.0         142.150.10.224  0.0.0.0         eth0
```

Question 2. [10 MARKS]

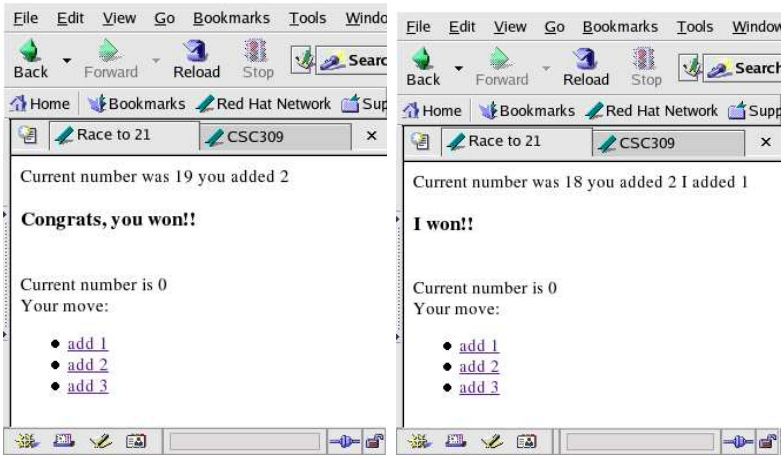
Create Race21 Java Servlet which plays Race to 21: You should make good use of session and/or context attributes as appropriate. and make good use of the Race21 class.

The *Player* plays against the *Server* with the Player moving first. The Server maintains the *current-Number* which stars at 0. During their turn, a participant adds one of 1, 2 or 3 to the current number. The participant that moves the currentNumber to 21 wins the game.

Below is a sequence of screens in a game: The Player visits
http://www.gamesite.com:8080/games/servlet/Race21Servlet (to get the left screen shot), next they click on the 'add 3' link, next they click on the add 2 link.



A Player and a Computer winning screen are shown below.



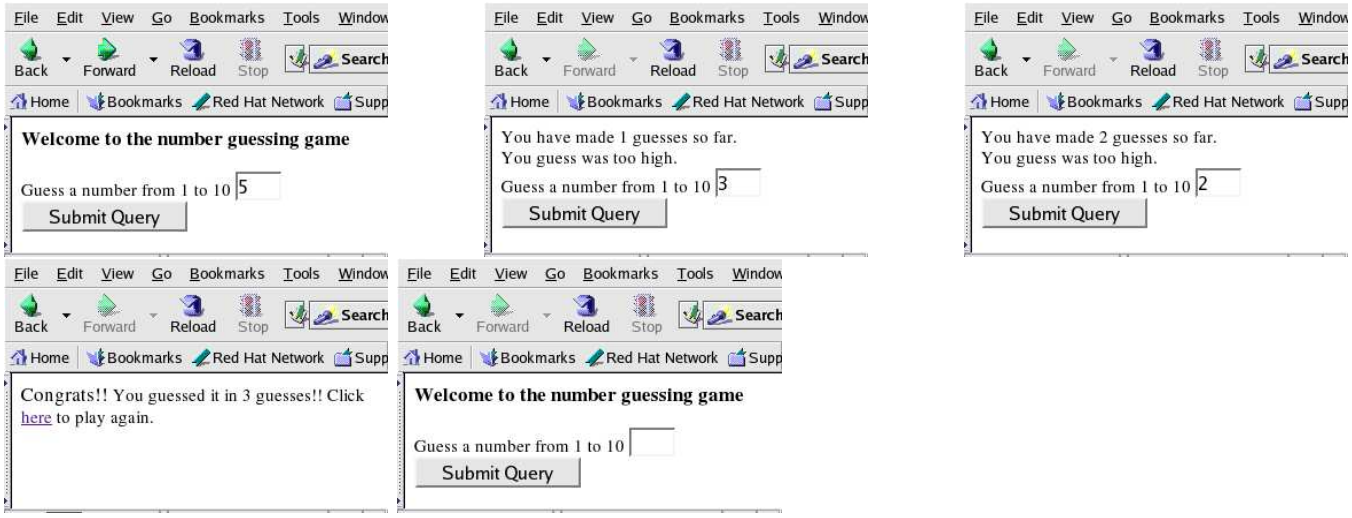
```
public class Race21 {
        public static int NOBODY=0, COMPUTER=-1, PLAYER=1;
        private int winner, currentNumber;

        public Race21(){ winner=NOBODY; currentNumber=0; }
        public void playerMove(int i){
                currentNumber+=i;
                updateWinner(PLAYER);
        }
        public int computerMove(){
                currentNumber+=1;
                updateWinner(COMPUTER);
                return 1;
        }
        public int getWinner(){ return winner; }
        public int getCurrentNumber(){ return currentNumber; }
        private void updateWinner(int potentialWinner){
                if(currentNumber==21)winner=potentialWinner;
        }
}
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Race21Servlet extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws IOException, ServletException
    {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
```

(continued...)

Question 3. [10 MARKS]

CGI/Perl: Create the perl CGI scripts described below.

- When a player visits **http://www.numberGuess.com/cgi-bin/startGuess.pl** a random number is chosen (via **my $numToGuess=1+int(rand(10));**) and stored as a cookie on the players browser.
- A users guess is actually processed at **http://www.numberGuess.com/cgi-bin/guess.pl**. This page submits back to itself, it advises the user as to whether the submitted guess is too high, too low or correct.
- The number of guesses made so far (for a particular number) is stored in a hidden input and displayed when the user guesses the servers chosen number.
- An example execution is shown below. Your implementation must follow this example. The top left and bottom right pages are the **result** of the execution of startGuess.pl. All other pages are the **result** of guess.pl.



(continued...)

Question 4. [14 MARKS]

Full marks only for **complete, clear, concise** answers. No obvious questions are left unanswered. Your answers here should indicate that, in very short order, you could perform the required task.
Consider again the numberGuess.com website and the scripts startGuess.pl and guess.pl. Describe how an outside user (with no access to source code) could **simply**...

- Determine whether any hidden variables are used in the application.

- Without using a browser, determine which cookies the application intends on storing on the browser and their values. Explain in detail!

- Fool the server into thinking that the chosen number is 50000, and that the user has made 2300 guesses so far and that the current guess is 17234. Explain in detail!

(continued...)

Question 5. [10 MARKS]

- What are POST and GET?

- Sid claims that cookies and sessions are the same thing. Explain both of these and so clear up any confusion Sid has about these concepts.

There is NO question on this page!

*[If you need extra space to answer a question, use the space below and indicate **clearly** the question number.]*

There is NO question on this page!

*[If you need extra space to answer a question, use the space below and indicate **clearly** the question number.]*

Total Marks = 54