Practice Final Exam

22 December 2011 from 11–1 in the lab

You have two hours, if you need it. Write your answers on this page, or additional blank sheets. Put your name at the top of each page. You may not use books, notes, computers, or other devices. You may use a calculator. You may leave when you have completed the exam.

1. For each statement below, fill in the blank with the *best* term from the following list. Some terms might be used more than once; some might not be used at all.

   • domain name • foreign key • frequency analysis • HTML • HTTP • IP address
   • password • primary key • public key • table

   (a) ______________________ is a language that specifies the content of web pages.

   (b) A(n) ______________________ is a numeric identifier (currently 32 bits) assigned to each host on the Internet.

   (c) The most common mechanism for authentication between humans and computers is a(n) ______________________ .

   (d) ______________________ is the technique used to decode simple forms of cryptography, such as single substitution.

2. This question is about the relative strength of two kinds of authentication mechanisms:

   (a) A bank PIN that uses a sequence of seven digits. (For example: 740 3472)

   (b) A password that uses five lower-case letters. (For example: `ahquo`)

   Do you have a sense about which mechanism is stronger? How many possible PINs/passwords are there in each case? How does this result affect the strength of the password?

3. The three tables below are a simplification of the database for a social networking web site, like Facebook. There is one main table, 'User', and two other tables that contain foreign keys to 'User'.

User:

ID*	Name	Birthday	Password hash
1	Dee Doe	1989/12/21	98246ef16a87c12407e5fada044f591e
2	Edward Cho	1990/11/19	1ca30cd59f0b566f9ef3a8208679585e
3	Francine Fuentes	1985/03/25	e5dbb7657f770fad038220f5c69d806c
4	Carl Carlson	1970/05/03	61aa5b6c78fa4e3636069347ae39df10
5	Alice Ann	1974/08/18	cf6a52053ff904bca9d96fd4e7740d7d
6	Bob Björk	1982/11/07	75e22f4965738386cbe02bca10d3120d

Friendship — indicates which users are friends with which other users:

User 1 (ref. User)	User 2 (ref. User)	Status	Date
1	2	approved	2005/12/20
1	4	approved	2010/03/24
1	5	approved	2007/05/06
1	6	approved	2010/03/08
2	3	approved	2006/11/01
2	4	approved	2011/08/03
3	4	approved	2008/09/04
4	5	requested	2009/08/04

Wall Messages — sent between users:

Sender (ref. User)	Receiver (ref. User)	Date/time	Message
1	2	2011/12/05 11:51	"Hey man!"
1	5	2011/12/05 16:40	"What r u doing tonite?"
2	1	2011/12/05 17:45	"Send me some tunez"
2	3	2011/12/05 21:18	"Love that pic, LOL"
4	2	2011/12/05 23:00	"This prof is trying my patience."
3	1	2011/12/06 00:05	"Ugh, tired"
2	4	2011/12/06 06:37	"You rock!"

(a) Which user is the youngest? ______________________

(b) The oldest friendship in the database is between which two users?

______________________ ______________________

(c) Name all the friends of Carl Carlson.

(d) Are there any wall messages between users who are *not* friends?

4. Why is requiring a *shared secret* a serious weakness of many cryptographic techniques, and how does a public key system address this weakness?

5. Below is a table of jobs that we must schedule on a batch operating system.

Job	**Run time**
J1	5 seconds
J2	4 seconds
J3	2 seconds
J4	4 seconds

(a) Create a time-line to illustrate the First-Come First-Served (FCFS) strategy. It should include the start/stop times of each job.

(b) Compute the average *turnaround* time of the four jobs using your FCFS time-line from the previous question.

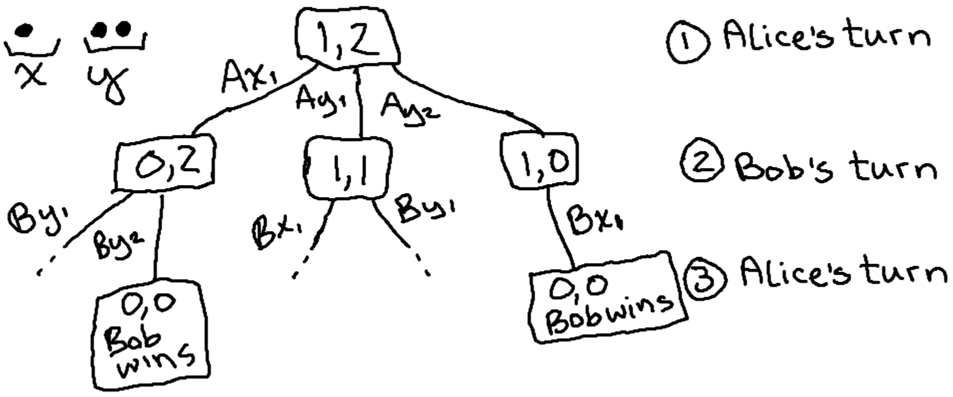
(c) Create a time-line to illustrate the Shortest Job Next (SJN) strategy. It should include the start/stop times of each job.

(d) Compute the average *turnaround* time of the four jobs using your SJN time-line from the previous question.

6. This question is about a strategy for an ancient game that is known in modern times as *Nim.* We start with several piles of stones; let's call the piles x, y, z. On each turn, a player can remove *any number* of stones from a *single* pile. The player to take the very last stone wins.

   We're going to investigate the search tree, as if we were programming a computer player. This is for an end-game, where only two piles remain, pile x with one stone, and pile y with two stones. So we represent that state as $1, 2$. We can represent moves as Ax_1 (which would mean Alice takes one stone from pile x) or By_2 (which would mean Bob takes two stones from pile y). It is Alice's turn.

   Below is a partial tree of moves for each player. Complete the tree to the end of the game, show who wins in each case, and then use those results to determine which is Alice's *best move* for turn #1.


x
y
1,2
Ax1
Ay1
Ay2
① Alice's turn
0,2
1,1
1,0
② Bob's turn
By1
By2
Bx1
By1
Bx1
0,0 Bob wins
0,0 Bobwins
③ Alice's turn