Name: ___________________________ Year: FY SO JR SR

Major: ___________________________ (fresh/soph put possible major if known)

Are you registered for COMP10 (circle one)?

YES I PROBABLY WILL I PROBABLY WON’T THANKS, JUST LOOKING

Why are you taking this course?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

What do you hope to gain from this course?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

How much (if any) experience have you had using computers?

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

________________________________________________________________________

(Note: if you have programmed, you should take comp11)

Thank you
<table>
<thead>
<tr>
<th>problem</th>
<th>sort 10 cards in increasing numerical order</th>
</tr>
</thead>
<tbody>
<tr>
<td>solution summary</td>
<td></td>
</tr>
<tr>
<td>procedure</td>
<td></td>
</tr>
<tr>
<td>notes</td>
<td></td>
</tr>
</tbody>
</table>
Topics: Introduction to Course

Approach: Discussion, Explanation, Discussion

Main Ideas:

1. Brief Introduction
   - Agenda
   - Teaching staff - who we are
   - Students - who they are

2. What is this course about?
   a. Summary -
      - How computers and the Internet work and (theory)
      - how to work computers and the Internet (practice)
   b. Organizing Idea
      - Problems and Procedures -
   c. Focus
      - Information - representation, storage, transmission, processing
   d. Method: WTP, WTS, HIW, LMTO
   e. Approach
      - Look at specific problems, study and create solutions
   f. Problem 1: How to get a bunch of pictures to a bunch of people?
      - compare 1809, 1909, 2009
      - WTS/HIW
   g. Details of 2009 solution - procedure and questions (the big picture)
      - transfer, storage, shrinking, making pages,
      - making dynamic pages, changing slides,
      - all happens automatically
   h. Examine some 2009 Solutions
   i. Beyond images: how could you send sound, text, smell, taste, ...

3. The course
   - A Language Course - Learn by doing and practice
     - Structure of class - lectures, labs, projects
     - No clear textbook - take notes, and ask
     - I have topics and goals, you can set examples
     - Sign up for labs, read website

4. From Problem to Procedure - sorting
   a. Computers sort things - by date, size, title, ...
   b. Problem: How to put a bunch of things in order
   c. Solution: you try it
   d. Write down your procedure so someone else use it (like a recipe)
   e. Discuss/Compare procedures
   f. What if you had 100 cards to sort?
   g. Next time: how could you build a machine to do this?

5. What next?
   a. Complete student info sheet
   b. Homework due next Wednesday
Course Content - Problems and Procedures

*Introduction to Digital Information* introduces basic concepts and skills about computers, the Internet, computer science, and computer programming. The content can be described as *Problems and Procedures*. We look at problems computer science addresses and the procedures computer scientists, and you, devise to solve those problems.

After completing COMP10-IDI, you should have a basic understanding of how computers and the Internet work and how to work computers and the Internet. Moreover, graduates of COMP 10 are prepared to continue studying computer science by taking COMP 11.

Course Approach - WTP, WTS, HIW, LMTO

In comp10, you will learn how to build a web-based slideshow. In doing so, you will learn how computers and the Internet work.

An automated system to publish and present a set of photos on a website requires many cooperating systems. You will learn how computers store, transmit, and process images and text. You will learn to manage files on a Unix/Linux server. You will learn to design web pages and to program pages to animate images and respond to user-controlled buttons and input. We study each of these systems and learn how they work and how to use them. The approach is to follow these four steps: What's the problem?, What's the solution?, How's it work?, Lemme Try One

Prerequisites: No computing, Some math, No computer

COMP10 is designed for people new to the field. We assume you use email and the Web, but we do not expect you to know how computers work, how to program, or any other ideas or skills in the field.

We do require basic competence in precalculus level math. You must know the basic ideas and skills of functions, variables, equations. A short placement test at the first lab will help us identify people who may need to take a math course first.

You do not need to own a computer to take this course. You can use any computer in the general labs on campus. Halligan hall is open 24 hours a day (you need to use your ID card to enter the building and labs after hours and on weekends.

What topics are covered?

- Basic operation and components of computers and the Internet.
- Working with files; transfer, storage, organization
- Image processing tools
- Automation using Unix scripts
- Hypertext Markup Language (HTML).
- Basics of algorithms and computer programming using Javascript.
- Closer look at computers and the Internet: hardware level, machine level, and higher levels.

Should I take COMP 10 or COMP 11?

If you want to take only one course that explains the basics of computers and the Internet, take COMP 10.

If you might want to take several computer science courses but have never had a course in computer programming, then take COMP 10. It will prepare you for COMP 11, which is the entry way for all advanced computer science courses. If you already know how to write simple computer programs (the computer language you know does not matter), then take COMP 11, not COMP 10.

Administrative Details

**People**

Instructor: Bruce Molay (molay at cs.tufts.edu)
Office Hours: Tue evening

Teaching Assistants:
Office Hours: click on the help link above

**Classes**

There are three meetings each week -- two lectures and one lab. Lectures are on Monday and Wednesday at 10:30 AM. Labs are on Tuesday. Classes explain ideas and study sample programs. Be prepared to take notes. All sample programs used in class will be stored on line, so you can retrieve them and examine and/or print them later.

**Labs**

There is a lab each week. Sign up for a lab using the form on the Getting Started page. At the lab you will work through projects similar to examples in class. You will turn in your lab work for a grade. Labs will be in Halligan 122.
**Homework**  Weekly assignments consist of short problems based on lecture and lab material or longer projects that require applying several ideas.

**Testing**  Several quizzes, two hour exams, two lab exams, and one final.

**Grading**  Assignments 30%, Labs 5%, Hour Exams 9% each, Lab Exams 9% each, Final 24%, Quizzes + Participation 5%

**Office Hours**  In Halligan Hall, hours and locations to be arranged.

**Texts**  The required text is: *A Balanced Introduction to Computer Science* by Reed. The 2nd edition is the current version, but the 1st edition is very similar. **Get the 2nd edition if you can.** Copies are on reserve.

Supplemental texts that may be helpful are *Your Unix: The Ultimate Guide, 2nd ed.* by Das; *JavaScript A Beginner’s Guide, 2nd ed.* by Pollock; *The Victorian Internet* is an optional, additional text. This book describes the invention and evolution of the telegraph system in the 1800’s. This short, inexpensive, well-written history shows how similar the telegraph system is to the modern Internet.

**Accounts**  You must obtain a Unix account on the eecs computers. Please visit [http://www.eecs.tufts.edu/~accounts](http://www.eecs.tufts.edu/~accounts) to set up or reset your ECE/CS Unix account.


**The Assignments and Time**

Think of COMP10 as both a culture class and a language class. You will study ideas and facts about how computers work and you will also learn to think and write in computer languages.

Some of the assignments, like those in history or culture classes, will require reading the text and answering short questions.

The majority of the work for the course, though, will require writing programs to solve problems. Learning a new language can be time-consuming and requires steady, regular effort. The programming assignments can take 10-15 hours (or more) per week. We offer lots of help during office hours and some via email.

**Do NOT fall behind on your homework!**  Learning to think and write in a computer language is like learning to think and write in a new human language -- you learn by doing and by regular practice. Just as you cannot expect to learn to speak a human language by listening to other people speak, you cannot learn a computer language by watching other people write programs.

Computer science, like foreign languages and math, cannot be crammed for at the last minute, so do not even try. We cannot stress this strongly enough. Imagine you are getting in shape for a race, running five miles each day. Say you get busy, and decide to postpone your practice, and just cram by running 35 miles on Sunday. Your body won’t let you. Similarly, if you decide to postpone work on a programming project until the last minute, your brain will lock up after several hours. This course counts for math credit; work in this course the way you should a math course.

**Getting Help and Collaboration**

If you need help with a programming project or the course material, please contact a TA or the instructor. It is best to visit either the TAs or your professor during their posted office hours. Failing that, try electronic mail. Tutors are available through Academic Resources. However, if you get a tutor, you should check with your instructor to make sure that your tutor is qualified.

Class and labs are times to discuss ideas, share solutions, and work with others to learn. We encourage students to work together during these times. If you prefer, you can work by yourself.

On the other hand, the policy on homeworks and projects for this course is that you must do all your work on your own unless you are explicitly told otherwise by your instructor. If you need help, you should see either a TA, your instructor or a tutor through Academic Resources. If you wish to discuss an assignment with others, you may discuss ideas but not share code. You may also discuss low-level details to resolve particular errors. You may not otherwise discuss or share the solutions to assignments via any medium.

You may use code from class or from the book in your solutions, but you must cite the source in the code, just as you would if you quoted something in a paper. If you receive assistance from a TA or from the instructor, please cite that source, also.

Each student is responsible for keeping his/her solutions private. For example, if you print out a listing of your project, you may not discard it in the user area as other students could retrieve it from the trash. Copying part of an assignment from another student, or being copied from, even if the assignment is later modified, will be treated as plagiarism. In addition, you are responsible for keeping your computer passwords secret. If anyone other than you is found using any of your computer accounts, whether they are in the class or not, you will be treated as if you received unauthorized help. Plagiarism or obtaining unauthorized help in this course will be treated in the same way as plagiarism in any other course, and can result in loss of credit, probation, suspension, or expulsion.
To prepare for COMP 10-IDI, you need to do the following:

1. Get the book
2. Get a CS Unix account
3. Sign up for a lab
4. Complete the student info form
5. Install software on your computer (if you have one)

Get the Book

The following book is required for this course:

- *A Balanced Introduction to Computer Science 2nd ed.*
  by David Reed, Pearson Prentice-Hall, Upper Saddle River, NJ

Reed's book is available in the Tufts bookstore, as well as from many web stores. Tisch has four copies on reserve. The first edition is pretty similar.

The following books are recommended for those interested in additional reference material on JavaScript and Unix. These books are not required for this course.

  by Pollock, McGraw-Hill, 2004

  By Das, McGraw-Hill, 2006

Get a CS Unix Account

If you are a returning student, your CS Unix account should already exist and should be the same as your trumpeter account.

If you do not have an account or have trouble logging in, try resetting your password online at http://www.eecs.tufts.edu/~accounts.

If none of these works, send email to staff@eecs.tufts.edu, include your Trumpeter login along with your request.

Sign up for a Lab

Go to http://www.cs.tufts.edu/~molay/c10.s09/website/forms/labs.html to tell us your lab choices.

Complete the Student Information Form

Go to http://www.cs.tufts.edu/~molay/c10.s09/website/forms/info.htm to tell us something about your background and interests.

If You Plan to Use Your Own Computer, Install Software

Note: You Do Not Need Your Own Computer for COMP 10. The computers in the labs in Halligan Hall have all the software you need for the course, and the computers in other Tufts computer labs have most of the software you need.

If you have your own Mac or Windows computer, please install the following programs:

Mozilla Firefox 2.0 or Later
For many operations, all web browsers (e.g., Netscape, Mozilla, Internet Explorer, Opera) operate the same way. But in some cases, they behave differently. Thus, for this course, we will standardize on Mozilla Firefox. If you use another browser to prepare an assignment and it does not work properly when we test it using Firefox, you will lose credit.

If you have a computer but do not have Firefox on it, then use any web browser and go to http://www.mozilla.com/firefox/. Then follow the instructions to download Firefox to your computer.

A Remote Login Program - ssh2

You use a remote login program to manage documents on your website. The ssh program provides a connection safe from electronic wiretaps.

- For Windows download ssh2 from http://www.cs.tufts.edu/comp/101D1/info/sshdownload.exe
- For Mac OSX Open a terminal using Hard Drive->Applications->Utilities->Terminal and then type ssh username@linux.cs.tufts.edu

A File Transfer Program

You use a file transfer program to move images, documents, and other files between your home machine and the web server.

- For Windows use ssh2 (see above), and click on the Window menu item to select New File Transfer.
- For Mac OSX get Fugu (http://rsug.itd.umich.edu/software/fugu).

A Plain Text Editor

All the work, except for image files, you submit for the course must be plain text. We cannot accept any Word documents, excel spreadsheets, or any non-text documents.

- Mac OSX can use TextEdit (which comes with OSX) or, for more power, download TextWrangler2 to use as a plain text editor for assignments.
- Windows users can use Notepad, which is under Programs->Accessories in the Start menu.
- Unix users can use pico, Emacs, XEmacs, VI, or a variety of other plain text editors. There should be at least one of these editors on your machine already.

Adobe PDF Reader

Much of the material for COMP10 will be available in the format called PDF. This format are easy to read and print. To view such files, you must have a PDF reader. You can get the "Adobe PDF Acrobat Reader" from Adobe's Acrobat Download Page.
<table>
<thead>
<tr>
<th>WEEK OF</th>
<th>TOPIC(S)</th>
<th>CHAP(S)</th>
<th>TESTS</th>
<th>ADMIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan 12</td>
<td>Intro</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan 19</td>
<td>Problems+Procedures</td>
<td></td>
<td></td>
<td>M1/19: no class</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W1/21: Monday schedule</td>
</tr>
<tr>
<td>Jan 26</td>
<td>Scratch</td>
<td></td>
<td></td>
<td>W1/28: last day to add</td>
</tr>
<tr>
<td>Feb 2</td>
<td>Scratch</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb 9</td>
<td>Unix Dirs/Files</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb 16</td>
<td>HTML</td>
<td>2</td>
<td></td>
<td>W2/18: last day to drop</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>M2/16: No class</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>R2/19: Monday schedule</td>
</tr>
<tr>
<td>Feb 23</td>
<td>Shell scripts</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 2</td>
<td>History,Algos</td>
<td>6,8</td>
<td>T3/3: lab exam</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>W3/4: exam 1</td>
<td></td>
</tr>
<tr>
<td>Mar 9</td>
<td>DROI</td>
<td>12</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Conditionals</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 16</td>
<td></td>
<td></td>
<td></td>
<td>Spring Break</td>
</tr>
<tr>
<td>Mar 23</td>
<td>JS1: Events, Elements</td>
<td>5,7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 30</td>
<td>JS2: Functions</td>
<td>9</td>
<td>W4/1: exam 2</td>
<td></td>
</tr>
<tr>
<td>Apr 6</td>
<td>JS3: Repetition</td>
<td>13</td>
<td>W4/1: last day to drop</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1styr)</td>
<td></td>
</tr>
<tr>
<td>Apr 13</td>
<td>Internet</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr 20</td>
<td>Internet</td>
<td></td>
<td></td>
<td>M4/20: no class</td>
</tr>
<tr>
<td>Apr 27</td>
<td></td>
<td></td>
<td>M4/27: lab exam</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>F5/1: final exam 7-9pm</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>M4/27: last day to withdraw</td>
</tr>
</tbody>
</table>

Note: This schedule is subject to change. Homework assignments are not listed here.
The Problem

Question: How to get a bunch of digital pictures to the computers of one or more friends?

The Procedure

Basic process of publishing pictures on the web

1. Take picture
2. Copy to computer
3. Copy to your public area on photo server
4. Friends and family can view the picture on your site.
How Does It Work?

Big Questions about the On-line Photo Album

How does data travel from place to place over the Net?

How do I make things happen when I click or type?

How do I arrange the pictures and text to make a nice page?

How can I reduce, crop, rotate images?

How are users, albums, pages, and images stored and organized?

How does a computer store images?

What goes on inside a computer?

And... How Can I Automate All These Actions??

Skills and Ideas

Elements of Computer Science Involved in an On-line Photo Album

network communication

page layout

buttons, forms, actions

unix programs

file transfer

remote login

digital images

computer internal structure

Programming: learn to tell computers what to do

file storage systems

web site

files and directories

image processing tools
Introduction

This assignment consists of two parts. One part is to figure out a problem posed on this sheet, the other is to read the first chapter in the text and answer some questions. Both are due on paper in class on January 21.

Part One: Analyzing Communication Systems

A small town far in the hills of Vermont has a school with children from 120 families. Sometimes school has to be closed because of snow. There is no radio station in town, so the principal is considering various communication systems to use to notify all 120 families. Your assignment is to analyze and compare these three systems.

The Three Systems

[1] The principal has a list of the telephone numbers of all 120 families. She calls every number, one by one.

[2] The principal calls the first person on the list. That person calls the second person on the list. The second person calls the third person on the list, etc.

[3] The principal calls the first two people on the list. Each of those two people calls two other people. And each person called calls two other people based on some chart devised by the school.

In all three cases, if nobody is at home at a number, the caller leaves a message if there is an answering machine. Of course, the people may be out of town and the message might not be received for several days.

Questions

1. For each of the three systems, how long will it take to tell all 120 families that school is closed? Make a reasonable estimate for how long each call takes. Explain your computation.

2. Imagine that exactly one family is out of town. What does this do to the effectiveness of each system? For each of the three systems: (a) how does the absence of one family affect other families, (b) what could the largest number of families not notified, and (c) what could be the smallest number of families not notified?

Your answer to question 2 shows that even one absent family can prevent other families from getting the news. Given that fact, answer questions 3 and 4.

3. Devise and describe a modification to each system that makes sure all present families are notified even if one or more families are not in town.

4. The principal likes your answers to question 3, but she still wants to know that all families that can be notified are notified. Devise and describe a further extension of the system so the principal knows when all present families are notified.

5. Consider modifying system [3] so that each person calls 3 other people, or 4 other people, or 5. Does making more calls shorten the time to notify everyone? Use numbers to support your argument.

6. Finally, consider 120 college students who like to go to parties. When any one of the 120 hears of a party, he or she wants to make sure all the other 119 people know about it. Do any of these three systems provide a good solution to this problem? Why or why not? Can you think of a different system to the party notification problem?

Part Two: Textbook Question

Based on your reading of Chapter 1 in the Balanced Introduction to Computer Science, write answers to the following Review Questions at the end of that chapter (pages 16-17): 1 through 11, 20 through 22.

Turning in Work

Turn in your answer on paper, word-processed or typed, not handwritten. You may include hand-drawn diagrams if you like. Put your name on the paper. Bring it to class.

If you discuss the problem with anyone and use any ideas they offer, you must cite them at the top of the page. We shall discuss team efforts soon.

No late work will be accepted. If you cannot finish all the problems, turn in what you have.