Course Outline: Math 4/5/7380, Fall 03

The web page for this course can be found at http://livetoad.org/. Please check there frequently for announcements, changes, solutions, and other goodies.

Introduction

Welcome to Discrete Structures and Analysis of Algorithms! This course is a foundation stone for theoretical computer science. Discrete — as opposed to the continuous — structures are the devices used to model problems in computer science. Algorithms are the language in which computational methods are discussed. Both of these predate the age of computers by centuries.

This course will not be a comprehensive introduction to discrete structures algorithms. Instead we will introduce ourselves to the subject by focusing on one particular application: encryption and privacy. The question we will try to address is what is information and where does it hide?

Encryption has a long history which in many ways shadows the general history of science and technology. In particular advances in the most abstract areas of mathematics soon find themselves in the most concrete applications. It is no coincidence that the National Security Agency is the world's largest employer of mathematicians.

The story of encryption is the story of the mythical Alice, Bob, and Eve. Alice and Bob are trying to communicate without letting Eve glean any information. They struggle back and forth over the centuries. Sometimes Alice and Bob succeed in hiding all of the information. Sometimes Eve breaks thru the barrier and clandestinely hears all they have to say. Too often the consequences of failure are absolutely disastrous for one party or the other.

To fully enjoy the spectacle of this epic battle, we will be forced to learn more and more mathematics. Quite surreptitiously, by the end of the semester we will have learned how to manipulate and understand finite fields. Until recently these were considered mathematical arcana, of no interest to the wider world. Now they are everywhere.

We will build computer models for finite fields, and we will build and test algorithms for their manipulation. The notion of an algorithm is sufficiently apart from other mathematical notions that you cannot really learn what they are until you build and test them yourself. This we will do with the computer programming language called *python*.

There is a difference between a computer program and an algorithm. A computer program implements (at least) one algorithm, but the abstract algorithm at its core is usually only part of the story. Building good programs requires an engineer's skill and attention to small details. We will try to ignore the engineering details as much as possible. In fact, it is best to strive for clarity and simplicity, at the possible expense of clever programming.

What makes computers valuable pedagogically is their complete stupidity. They do exactly what explicitly you tell them to do, nothing more and nothing less. They cannot and will not fill in the details for you, clear up ambiguities in your instructions, or read your mind. If a computer does not do what you expect it to do then what you said was not what you meant. We will strive to say exactly what we mean.

Python

I do not assume you know anything about computer programming. In particular I do not assume you know the language python. However I know from experience that if you put time and effort into learning python then you will be fluent rather quickly. I have structured the course so that we will not be learning many new mathematical skills or concepts until you have had time to become used to python. Plan to be spend a lot of time in front of a computer typing python, experimenting, making mistakes, and above all learning. Do so now.

I expect you to write correct python on all of your homework and exam problems. You will lose a significant number of points for incorrect python syntax. Python is perhaps the easiest language to learn, and so the only excuse for not writing correct python is failure to put in the time to learn it. Start now.

Python is available in the Math Dept computer lab, UH 1000. If you have your own computer then you should take the time to download and install python yourself. It is free. It is easy. Go to http://python.org/. The latest version is 2.3.

Quizzes

There will be a reading assignment nearly every day. At the beginning of the class we will have a short, 5-point quiz based on the assigned reading. I will post both the reading assignments and the quiz answers. Your 20 best quiz scores will count towards your final grade. I will not give make-up quizzes under any circumstances. If you miss a quiz then that will be one of the scores you drop.

Assignments

There will be 15 homework assignments, roughly one per week, each worth 10 points. I will post the assignments and due dates. I will not accept homework past the due date, unless you have a documented excuse, for example from a doctor. Your homework must be neat and show all work. Use complete sentences. You will not get credit for a solution if you are vague or if you omit important details. When you turn in your homework fold the papers lengthwise and write on the outside

your name, Math 4380, Fall 03, assignment number, due date

Exams

We will have two exams, a 100-point midterm exam and a 150-point comprehensive final exam. The exam dates are listed on the calendar below. The exam questions will be similar to those found on the quizzes and homework assignments. Some may require proof. Some may require that you write python code. If you are an undergrad then you will be given some choice of problems. If you are a grad student then on each exam you will have considerably less choice.

I will give make-up exams only in case of a documented emergency, such as illness or a funeral. If you are sick the day of the exam then you must call or email that same day if you expect to be able to make up the exam. Otherwise you must arrange for a make-up exam ahead of time. If I am not in my office then you can leave a voice mail message. If you fail to show up for an exam and do not contact me about it until afterwords then you will not be able to make up that exam — you will get a 0 for that exam.

Grades

Your final grade will be determined from the distribution of total points earned. I will post the class histogram, and this should give you a clear idea of where you stand. Historically 85% of total points earns an A; 75% earns a B; and 65% earns a C. However these are just historical observations, not rigid targets.

If you want me to post your grades under a nickname then bring me a 3×5 card with your name, an email address, and the nickname you want to use — preferably something not obvious!

If you stop attending class then I will give you an IW grade. There are two points during the semester for submitting IW grades: the 4th and 10th weeks. After the 10th week an IW grade is impossible, so if you stop attending after this point then you will get an F.

Office hours

My office is UH 4080e. The phone number is 419 530 2975. My email address is paul.hewitt at utoledo.edu. My office hours for this class are Monday and Wednesday before class. At these times you can call or stop by without an appointment and I am sure to be there. I am also available at other times, but for these you must make an appointment. Feel free to ask for appointments at other times if you cannot make it to my regular office hours. If you call me when I am not in my office then you can leave a voice mail message and I will get back to you as soon as I can.

Calendar

Labor Day	Mon	2 Sep
Exam 1	Wed	15 Oct
Last Day to Withdraw	Fri	17 Oct
Fall Break	Mon–Tue	20–21 Oct
Veteran's Day	Tue	11 Nov
Thanksgiving Break	Wed–Fri	26–28 Nov
Exam 2	Wed	17 Dec, 19:30–21:30