

CMPSCI 145

Review for Final Exam

Professor William T. Verts

Here are some notes on the take-home final and the upcoming final's week final exam. The take-home portion of the final had led to some questions in the past, so I thought I'd address a couple of things that might lead to some confusion.

TAKE HOME FINAL

The take-home final is optional. Here are your choices:

(1) Do nothing. Show up at the regular final exam, which will be worth 80 points. Your score out of 80 will be scaled by 1.25 to normalize it to 100 points.

(2) Do the take-home final, which is worth 20 points. Bring your essay to the regular final exam. Do the regular final exam, which is worth 80 points. At the end of the final make a decision to either include the take-home or not. If you decide to INCLUDE the take-home, the score on the take-home (out of 20) will be added to your regular final exam score (out of 80). If you decide to OMIT the take-home, your final exam score will be computed as in option 1: scaled by 1.25 to normalize it to 100 points.

SPECIFICS ABOUT THE STORIES

Please remember that these stories were written some 50 years ago, when computers were large, incredibly expensive, notoriously unreliable, and horribly fragile devices.

What computing technologies would have been in use back then that we've talked about in class? Would the authors, as good as they were at future-prediction, have foreseen the pervasive use of the miniature computing devices we use today? How does temperature affect computing technologies? What references to solid-state computing and cryogenics are made (both stories)? How does the speed-of-light and other physical characteristics of matter factor in?

In *Crusade*, did the probes actually return to the homeworld? How would the story be different if the probes did/didn't return? Are "several quadrillion bits" enough to convey the information supposedly returned by the probes? How does the homeworld interpret evolved "biological" computers?

In *Machine that Won the War*, how big is the MultiVac? How is the size related to the technology with which it was built? Is this something reasonable to consider today? If yes, then why? If no, then how would we do it now?

FINAL'S WEEK FINAL

If you have an exam conflict, please let me know through the on-line form and we can reschedule you for some other time during final's week, at our mutual convenience.

The exam is open-book, open notes (anything on paper is OK to bring in). Simple calculators are allowed (no graphing calculators, no calculators that have built-in base conversion functions, and no cell phone calculators, please). No other electronics permitted (cell phones, computers, PDAs, iPods, etc.). I will try very hard to make the exam short enough to do in the 2-hour exam time, but when you get the exam please go through it once to identify the easy problems and get them out of the way quickly. If you run short on time (which may happen), I do not want you to have a bunch of easy problems left unanswered.

I know that there is a lot of material here, and obviously I cannot ask detailed questions about each topic. Here are the high spots. I'll try to ask representative questions from each topic group and still bring it in under the time limit. Feel free to email me with questions over the next few days; if there is some topic that comes up I feel to be of benefit to the entire group I'll write it up and send it out as a broadcast.

The exam will mostly cover material from the second midterm onwards, but you should review the earlier material as well.

APPROPRIATE SECTIONS FROM THE BOOKS

The information below is from the on-line version of ***Representing, Storing, and Retrieving Information*** on the class Web site.

Sections I would like you to study are listed with "YES". Sections marked with "SKIM" contain items we only touched on or were early in the semester, and were mostly covered on the midterms. Sections marked with "NO" or formatted with ~~strike through~~ will definitely *not* be on the exam.

Where appropriate, there are corroborative pages from the ***Computer Science Companion*** that you should examine as well. These are marked as SEE ALSO COMPANION.

SECTION 1 - INTRODUCTION

Chapter 1 - Introduction (SKIM)

This introductory chapter covers the basic philosophy of the course: why different representations are important, and why we might choose one over the other.

SECTION 2 - HISTORICAL TECHNIQUES

Chapter 2 - Analog Mechanical Techniques (SKIM)

Remember how to compute gear ratios and how that relates to multiplication and division. Know how the linear and logarithmic Slide Rules work. Have a basic understanding of how simple Electrical Telephones work.

Chapter 3 - Digital Mechanical Techniques (SKIM)

Have a passing familiarity with the Abacus and Comptometer. Know how to use 9's and 10's Complement Arithmetic to perform subtraction via adding. Have familiarity with Key-Sort Cards and how card-based representations differ from paper-tape representations.

Chapter 4 - Electrical and Electronic Devices (SKIM)

Skim the material on electronics (Analog Division via Resistors), but know the differences between Analog Functional Elements (ENIAC) from the homework, and Digital Functional Elements (Gates). Skim only Analog-to-Digital and Digital-to-Analog conversions. **SEE ALSO COMPANION CHAPTER 6 PAGES 113-116, CHAPTER 5 PAGES 93-104.**

SECTION 3 - NUMBERS AND ALGEBRA

Chapter 5 - Integer Representations (YES)

Know this chapter solidly. Concentrate on Unsigned Integers and 2's Complement Signed Integers, but also have some familiarity with Sign and Magnitude, 1's Complement, and Gray Codes. You should also know Binary Coded Decimal (BCD), and be familiar with Excess-Three (XS3). **SEE ALSO COMPANION CHAPTER 4, and CHAPTER 3 PAGES 36-41.**

Chapter 6 - Real Number Representations (YES)

In this chapter you should know Floating-Point representations well, but also have some familiarity with Fixed-Point representations. Know about how Scientific Notation relates to Floating-Point. Know about the Sign bit, know the difference between the True Exponent and the Biased Exponent, and know how the Biased Exponent and Mantissa are constructed for Half, Single, and Double-Precision numbers, and be able to generalize. Be familiar with the differences between normalized numbers, denormalized numbers, infinity, and NaN. Be able to extend the methods of floating-point to any desired precision of my choice. **SEE ALSO COMPANION CHAPTER 4.** From the Companion: be able to discuss round-off errors, rounding, and interval arithmetic.

Chapter 7 - Symbolic Computation (SKIM)

We discussed briefly the tree form of equations and basic transformations of one tree into another.

Chapter 8 - Equations of Lines and Curves (YES)

Know about the Slope-Intercept and Parametric form of Lines and Parabolas. You should have some appreciation for how LaGrange interpolation applies to Parametric Cubics and Bezier Curves. **SEE ALSO COMPANION CHAPTER 3 PAGES 64-67.**

Chapter 9 - The Frequency Domain (NO)

~~Know that Fourier Analysis decomposes a periodic signal into a fundamental sine wave plus harmonics. If we discard high frequency information, and resynthesize a signal from what remains, the result is an ****approximation**** to what we had at the beginning, but contains less information overall. This is partly what is done when compressing an audio signal into .MP3 format.~~

Chapter 10 - Geometric Issues (SKIM)

Understand the process of ray-tracing, and why spheres are particularly nice objects to ray-trace (it is computationally easy to determine the distance between two spheres and between a sphere and a line/ray), as are triangles (always planar).

SECTION 4 - FILE FORMATS

Chapter 11 - Text Issues (SKIM)

Know that text characters are represented by bytes in ASCII or multiple-bytes in Unicode, and that the order of characters in a character set (such as ASCII) can allow us to easily perform tasks such as capitalization or encryption.

Chapter 12 - Graphics Issues (SKIM)

Know how basic color perception occurs. ~~In converting a 24-bit true-color image into a format with reduced color depth, know the difference between posterization and dithering.~~ Also, know that conversion losses when saving line-art or cartoons to .JPG format results in color "halos" at what should be sharp color boundaries. **SEE ALSO COMPANION CHAPTER 17 PAGES 281-284.**

Chapter 13 - Audio Issues (NO)

~~Understand how basic .WAV files are constructed, with respect to the number of audio channels, number bytes per sample (amplitude measurement), and the number of samples per second. Know about how Nyquist's work relates to the selection of the number of samples per second. Know how people approach the problem of compression to a format such as .MP3.~~ **SEE ALSO COMPANION CHAPTER 18.**

Chapter 14 - Video Issues (NO)

~~Know how the size of a video file is related to the size of a frame, the number of bytes per pixel, and the number of frames per second, as well as some of the compression techniques that are used to reduce the file size.~~

SECTION 5 - SEARCHING AND SORTING

Chapter 15 - Simple Searching and Sorting (YES)

Know the difference between linear search and binary search, and know how to perform each of the self-organizing list options. Know that linear searches are $O(N)$ in the worst case, even when enhanced with self-organizing techniques that can improve search time for some data sets, that binary searches are $O(\log_2(N))$, and that sorting is between $O(N \times \log_2(N))$ for the best sorting techniques (heap sort, merge sort, quick sort) and $O(N^2)$ for the worst techniques (bubble sort, exchange sort, etc.). Study hashing and Soundex. There may be an exam problem concerning the SoundEx algorithm. **SEE ALSO COMPANION PAGE 27.**

Chapter 16 – Databases (YES)

Read and understand how real databases represent the records in memory and how searches work. You should understand the differences between sequential access, random access, and hierarchical access, and what advantages and disadvantages each offers relative to the others.

SECTION 6 - LANGUAGES

Chapter 17 - Markup Languages (NO)

~~SKIP. Read if you are interested in Web design.~~

Chapter 18 - Computer Languages (NO)

~~SKIP. Read if you want to know about programming.~~

Chapter 19 - Natural Languages (NO)

~~SKIP. Read if you are interested in some of the early research into understanding of human languages by computer.~~

SECTION 7 - INFORMATION THEORY

Chapter 20 - Basic Information Theory (YES)

Know the differences (advantages and disadvantages) between serial and parallel data transmission. Know about parity bits and check sums, and how voting out bad bits can happen in the case of sending the same bit three or more times. Know that if you send a bit twice you can detect an error but cannot correct it. Know how to encode and decode a message. Know how Hamming Distances are computed. **SEE ALSO COMPANION.**

Chapter 21 – Compression (YES)

Understand the difference between lossy and lossless compression. Know the basic techniques such as run-length encoding and delta modulation. Know in detail how the Huffman coding compression technique operates. Know in general terms the difference between a two-pass approach such as Huffman and a one-pass approach such as LZW. **SEE ALSO COMPANION CHAPTER 19.**

Chapter 22 - Codes & Ciphers (YES)

Read and understand the "real world" example of coding using the thermistor and BASIC Stamp. Be aware of other coding techniques such as Morse and Braille (and binary, of course). Know in detail how single-key encryption works (in particular, you must understand that knowing the key to encrypting a plaintext message also gives you the key to decrypting the cipher-text). Know how public-key (double-key) encryption is used to both encrypt and to sign a message. Be able to encrypt-decrypt a message using a Caesar Cipher using the wheels on the class Web site. Know how steganography is similar to and different from traditional encryption, and know how to hide a message in images as we did in one of the assignments/demonstrations. **SEE ALSO COMPANION CHAPTER 19.**

Please go to the class site and download and print the Caesar Cipher wheels. I recommend printing them at 125% scale to fit the paper, but you may need to play with this a bit. Cut out the two wheels and bring them to the exam. There will be a Caesar Cipher problem on the final.

http://www.cs.umass.edu/~verts/compsci145/CaesarCipher/Dual_Rings.gif

SOME GENERAL QUESTIONS

What is the difference between an analog device and a digital device? What general characteristics of each type lead you to decide whether a particular device is analog or digital?

How do you use 9's and 10's complement arithmetic to perform subtraction using a machine capable only of addition? How does XS3 tie in here?

How do you use 1's and 2's complement binary arithmetic to perform subtraction using addition? How do you convert between decimal (base 10) and any of the listed binary forms? What are the problems with each of the signed binary forms (Sign and Magnitude, 1's Complement, 2's Complement) and where on the number wheel do those problems occur?

What are the advantages and disadvantages of the BCD and XS3 forms? What kinds of problems do those representations attempt to solve? How do you perform an addition in BCD or XS3?

How do you convert a decimal (base 10) number with a fraction into rational, true binary, binary scientific notation, and then into single or double precision? How do you convert a single precision floating-point number into decimal? What is a Denormal and why do we use them? What happens when we attempt to store a number which is irrational or a repeating rational in binary (such as the binary value of one-tenth) into a floating-point number format?

What are the advantages and disadvantages of the implicit form versus the parametric form of a line? Given a set of points, how do we generate the parametric form of a line (two points), parabola (three points), or cubic or Bezier curve (four points)? What are the rules about extending curves to three, four, or more dimensions?