

**CMPSCI 145  
SPRING 2015  
REVIEW SHEET FOR MIDTERM EXAM #2  
Professor William T. Verts**

The exam is open-book, open notes. Anything on paper is OK to bring in. Simple calculators are allowed. No calculators that have built-in base conversion functions, and no cell phone calculators, please. Graphing calculators are allowed but discouraged. No other electronics permitted (cell phones, computers, PDAs, iPods, iPads, etc.). I will try very hard to make the exam short enough to do in the 50-minute class 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, I do not want you to have a bunch of easy problems left unanswered. Feel free to email me with questions over the next few days; if there are topics I feel to be of benefit to the entire group I'll write them up and send them out in an email broadcast.

***Review Questions***

How do you convert between decimal (base 10) and any of the binary forms we covered? 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? How is the act of shifting a number left or right in binary equivalent to multiplication or division by two?

How do you use 1's and 2's complement binary arithmetic to perform subtraction using addition? How is 1's complement similar to 9's complement, and how is 2's complement similar to 10's complement?

What are the advantages and disadvantages of the BCD and XS3 forms? How does BCD and XS3 tie in with 9's and 10's complement arithmetic? What kinds of problems do those representations attempt to solve? How do you perform an addition in BCD or XS3? What advantages does the Gray Code binary representation have over regular binary? When can Gray code be used to its advantage?

How do you convert a decimal (base 10) number with a fraction into rational, true binary, binary scientific notation, and then into quarter, half, single or double precision? How is binary scientific notation similar to and different from traditional decimal scientific notation? How do you convert a quarter-precision, half-precision, single-precision or double-precision floating-point number into decimal? What is a Denormal and why do we use them? How do we represent Infinity? What happens when we add 1 to Infinity? What is a NaN and how are the two types of NaN used? Why do we drop the leading 1 bit in the mantissa, except for Denormals? What are the common binary patterns for zero, the smallest non-zero Denormal, the largest Denormal, the smallest normalized number, and the largest normalized number for any of the listed precisions? Why are exponents biased? What are the biases for different widths of exponents? Can we have positive and negative zero in floating point? What happens when we attempt to store a number that is irrational or a repeating rational in binary (such as the binary value of one-tenth) into a floating-point number format?

Not in the book, but covered in class: What is interval arithmetic and how can it lead us to knowing whether or not to trust a computation?

How do we use symbolic computation to avoid many of the problems with floating-point arithmetic? How is a rule such as “Replace  $(X + 0)$  with  $X$ ” or “Replace  $(X / X)$  with  $1$ ” or “Replace  $\text{Cos}(\pi / 2)$  with  $0$ ” applied? How do these rules help us with the limitations of floating-point? How would you resolve the conflict between the two rules “Replace  $(X ^ 0)$  with  $1$ ” and “Replace  $(0 ^ X)$  with  $0$ ” when  $X=0$ ? How do you develop new rules to include? How do you reason about a tree-based structure for a symbolic expression?

What advantages does the parametric form of a line or curve have over the implicit form? How would you use Lagrange interpolation to pass a curve through some number of points? Can a curve go through more than three dimensions? How can you join two or more Bézier Curves or Quadratic Splines together end-to-end so that each one blends smoothly into its neighbor? How is DeCasteljau’s algorithm adaptive to the shape of the curve when plotting it on screen? How does the DeCasteljau approach use recursion?

What information do we need to have to know everything about a circle on a plane or a sphere in space? What about a triangle in space, or a cylinder? How do I know if two circles or spheres overlap, touch at one point, or are completely separate? How does ray-tracing work? What happens to rays cast from the eye into the model when objects are reflective or translucent? How do light sources affect rays to create shadows?

What are the differences between vector graphics and raster graphics? What are the advantages and disadvantages of each approach? What are the primary colors of transmitted light? How many bits available are for each of the three primary colors in each full-color pixel commonly used today? How many possible colors are there in that representation? What happens to a full-color bitmap when saved as a .JPG, .GIF, .BMP, or .PNG file? What is a palette? What is dithering? What is the difference between lossy and lossless compression? What is transparency in the context of .GIF or .PNG files? What is the trade-off between image quality and file size with .JPG files? What is sub-pixel sampling, and how can it be used to get a finer-than-pixel resolution for black text on a white screen? What is aliasing? What is anti-aliasing? What are the differences between a bitmapped image and an image in the .SVG format? When would I use each one?

Think about the graph drawing program and the sphere-tracing program I demonstrated in class. In each case, what are the differences between the representation of the *description* of the image and the image itself? When would I use each one?

What is a linear search? How do the various self-organizing list techniques aid in search time? What is a binary search? What has to happen before a binary search can be performed? What is “big-O” notation, and what is meant by  $O(1)$ ,  $O(N)$ ,  $O(\log_2(N))$ , etc.? What is sorting? How fast can sorting be performed?