CMPSCI 145 MIDTERM #2

Solution Key

SPRING 2018

April 13, 2018

Professor William T. Verts

<1> 10 Points – Answer 10 of the following problems (1 point each). Answer more than 10 for extra credit. Scoring will be +1 for each correct answer (+½ for partial credit), -1 for each incorrect answer, and 0 for blank answers. Your score will be the sum, but will not go below zero. (For example: 10 correct and 5 incorrect will give a total score of 5.)

131	What is the decimal value of the 8-bit ***unsigned*** binary number **10000011** ?
-3	What is the decimal value of the 8-bit ***sign and magnitude*** binary number **10000011** ?
-124	What is the decimal value of the 8-bit ***one's complement signed*** binary number **10000011** ?
-125	What is the decimal value of the 8-bit ***two's complement signed*** binary number **10000011** ?
83	What is the decimal value of the 2-digit ***BCD*** binary number **10000011** ?
50	What is the decimal value of the 2-digit ***XS3*** binary number **10000011** ?
11000001	What is the binary sum of **10000011** and **00111110**?
0001 0010 0001	What is the BCD sum of **10000011** and **00111000**? (You may need more than 8 bits.) Show the result in binary.
Interval Arithmetic	What approach keeps two numbers for each computation, one of which rounds results down and the other rounds results up?
Bézier Curve	What type of cubic curve is tangent at each endpoint to the curve through that endpoint and its corresponding control point?
Ray Tracing	What technique allows us to generate synthetic scenes by passing a line from the eye through each pixel into a 3D model?
Sorted	How must a list of items be configured so that a binary search on that list is possible?
Sequential Access	One of two traditional methods for accessing a database is called "random access". What is the other method?
Hierarchical Access	What is the third method of database access, as used in image pyramids?
C	Show how to symbolically reduce the following expression: **(C + (B – ((D/D) * B) + (A * (E – E))))**

<2> 25 Points – We are going to invent a new floating-point format, called "Strange-Precision" which is exactly fourteen bits in length. There is a sign bit, six bits for the biased exponent, and seven bits for the mantissa, and all coding rules are the same as single, double, half, and quarter precision.

A. (1 Point) What is the value of the bias for this format? $2^{6-1}-1 = 2^5-1 =$ **31**

B. (24 Points – 3 Points each – 1 point for every 5 bit errors) Fill in **all** the boxes below with 0 and 1 bits to form the indicated values. Leave no box blank.

SIGN	BIASED EXPONENT	MANTISSA	
0	1 1 1 1 1 1	0 0 0 0 0 0 0	Positive Infinity
0	1 1 1 1 1 1	1 X X X X X X	Quiet NaN (X can be either 0 or 1 or X)
0	1 1 1 1 1 0	1 1 1 1 1 1 1	Largest Positive Normalized Number
1	0 1 1 1 1 1	0 0 0 0 0 0 0	-1.0
0	0 0 0 0 0 1	0 0 0 0 0 0 0	Smallest Positive Normalized Number
0	0 0 0 0 0 0	1 1 1 1 1 1 1	Largest Positive DeNormal
0	0 0 0 0 0 0	0 0 0 0 0 0 1	Smallest Positive Non-Zero DeNormal
0	0 0 0 0 0 0	0 0 0 0 0 0 0	Zero

<3> 20 Points – Convert the number **13.1** into Strange Precision.

A. (3 Points) What is 13 in binary?

1101

B. (4 Points) What is 0.1 in binary (10 bits to the right of the decimal point)?

0.0001100110011001100110011…

-1 for fewer than 10 fraction bits.

C. (2 Points) What is the True Binary of 13.1 (10 bits to the right of the decimal)?

1101.0001100110011001100110011…

-1 for fewer than 10 fraction bits.

D. (4 Points) What is the Binary Scientific form of the answer in Part C?

1.1010001100110011001100110011… × 2^3

-1 for fewer than 10 fraction bits.

E. (2 Points) What is the Decimal Value of the Biased Exponent?

31 + 3 = **34**

F. (5 Points) Fill in the Strange Precision number below with the correct bits. If you need to round the fraction to fit, use "Round to Nearest, Round Half Away from Zero" rounding.

0	1	0	0	0	1	0	1	0	1	0	0	1	0

SIGN **BIASED EXPONENT** **MANTISSA**

1 point for sign bit, 2 points for exponent, 2 points for mantissa. Note that the number in binary is 0-100010-1010001 when the fraction is *truncated* (round towards zero), but because the first discarded bit is a 1 we round up the last bit, which carries into the next to the last bit. Subtract 1 point for truncating instead of rounding.

<4> 25 Points – There are two 3D points <-3,2,0> and <6,4,9>. What are the parametric equations in *x*, *y*, and *z* for the straight line that passes through the first point when *t=0* and through the second point when *t=1*?

A. (12 Points – 2 Points each):

x(t) = __*9*__ ***t*** + __*-3*__

y(t) = __*2*__ ***t*** + __*2*__

z(t) = __*9*__ ***t*** + __*0*__

B. (3 Points – 1 point each): What are the coordinates of the 3D point when ***t*** = +½?

< __**1½**__ , __**3**__ , __**4½**__ >

C. (4 Points – 1 point each, 1 free point for any answer): What are the coordinates of the 3D point when ***t*** = -½?

< __**-7½**__ , __**1**__ , __**-4½**__ >

D. (6 Points – 1 Point each):

Set up a LaGrange equation for a straight line to go through two points P0 and P1, where the line hits P0 when t = -1 and hits P1 when t = +2.

$$\frac{(\,t\ -\ \underline{+2}\,)}{(\,\underline{-1}\ -\ \underline{+2}\,)} P_0 + \frac{(\,t\ -\ \underline{-1}\,)}{(\,\underline{+2}\ -\ \underline{-1}\,)} P_1$$

<5> 20 Points – Searching lists.

Here is a list of names, in unsorted order. Show the list after each search, using the **Promote-One-Slot** self-organizing linear list technique (the front of the list is at the left):

Tom Fred Carol John Sam Mary Bob

A. (2 Points) After searching for Carol:

Tom **Carol** **Fred** John Sam Mary Bob

B. (2 Points) After searching for John:

Tom Carol **John** **Fred** Sam Mary Bob

C. (2 Points) After searching for Carol again:

Carol **Tom** John Fred Sam Mary Bob

D. (2 Points) What is the worst-case search time for this list using linear search? (That is, what is the maximum number of comparisons that must be made to find an item in this particular list, or determine that the item is not present?)

O(N) = 7 comparisons

E. (3 Points) If the list were sorted, what is the worst-case search time for a binary search for this list? (That is, what is the maximum number of comparisons that must be made to find an item in this particular list, or determine that the item is not present?)

$O(\log_2(N))$ = 3 comparisons

F. (2 Points) What is the worst-case search time for a hash function that uses a list of these names?

O(1) = 1 comparison

G. (2 Points) What is it called if two names hash to the same index?

Collision

H. (5 Points) Short answer: Explain how the American Soundex code is a hash function.

American Soundex reduces a surname to a code that represents a particular file folder in which to place the documentation for that person (unlike most hashing techniques, Soundex *expects* collisions so that related people go to the same folder). (Accept anything reasonable.)