COSC3319 Data Structures Test 3 MWF Spring 2017 Burris
(You may not use a calculator on this test!)

Answer 4 of the following question including question 1. *A professional should be able to answer every question*! Number your questions on the answer sheets in ascending numeric order from 1 through 6 inclusive. Clearly write “Delete” on the answer sheets by the 2 questions you do not wish graded. Leave at least a one-inch margin at the top of every page. You may answer multiple questions the same page as long as they are in ascending order. Do not write on the back of pages. Staple your answer sheets (in ascending numeric order) on top of the test in the upper left hand corner. Write your name in the upper right hand corner of the answer sheets (first page). Turn the stapled bundle over and write your name in the upper right hand corner on the back of the test. A five point Road Map Fee (RMF) will be deducted for each instance of failure to follow instructions. You will not receive credit for material I cannot read or that is obstructed from my view. **Please note some questions require substantially more time to answer than others!**

Warning: There are no “short” answers on this test. Tell me everything germane to the topic. Your performance is being compared to all other members of the class.

1) Assume a hash table of size eight using the random probe technique to handle collisions. The contents of the hash table were initialized to spaces prior to inserting and deleting records. Utilize “&” appropriately in completing your answer. The random number generator generates the following sequence of random numbers after initialization each time it is utilized: 5, 2, 7, 6, 3, 1, and 4. The initial table for part “A” appears below. Be sure to copy the results of part “A” into the table for part “B” prior to performing the part “B” operations.

A) First exhibit the contents of the table on the answer sheet as Table I after performing all of the following operations. Insert Tom at 3, Mary at 4, Bob at 6, Sue at 3, Sam at 6 and Jill at 2. Delete Sue.

B) Second insert Anna at 3 and delete Bob from the table from part A. Exhibit the results as Table II on the answer sheet.

1	
2	
3	
4	
5	
6	
7	
8	

Table I

1	
2	
3	
4	
5	
6	
7	
8	

Table II

2) Discuss the division remainder method for calculating hash addresses in detail including advantages and disadvantages. Your discussion should include but not be limited all restrictions associated with the table size, selecting the bits for the key, limitations imposed by different data types (e.g., character, numeric, etcetera), and utilizing keys whose size exceeds the number of bits in an integer. Do not consider collisions.

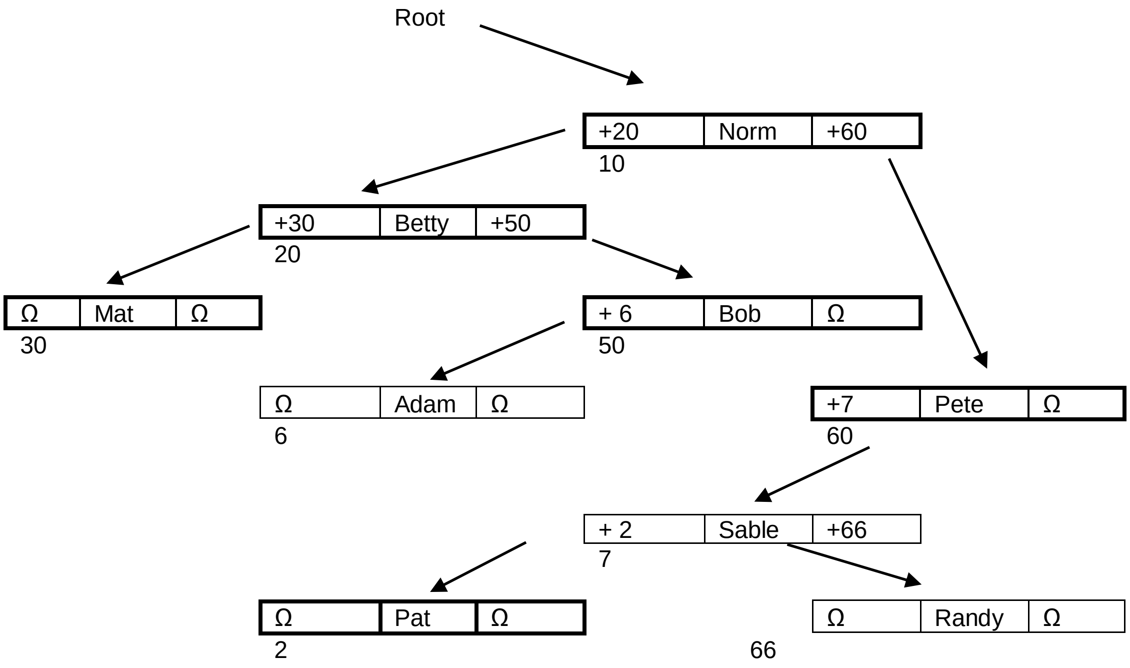
3) Evaluate the quality of the following hash function for keys from 4 to 16 characters in length. Assume the number of bits in an integer is 32. You must be very specific to receive credit. Assume a main memory hash table of size 128. The hash address is to be calculated using folding combined with the division remainder technique. The hash address must be computed by ((adding characters 1-2 of the key to characters 6-7 of the key)* 256) plus the 13th character of the key) modulo 128. *All characters must be treated as positive integers in all steps of the calculation*! Use the remainder as the hash address. For example, it the string contains the value "ABCDEFGHIJK45678" then the hash address would be HA ← (("AB" + "FG") * 256) + 5) modulo 128.

4) Write an iterative algorithm to traverse a binary tree pointed to by "Root" in preorder using a stack. You may use "=>" and "<=" to indicate push and pop with respect to a stack.

5) First print the contents of the following tree as you traverse it using the following recursive algorithm.

Algorithm Mystery Traversal:
1) Traverse the right subtree of the current node.
2) Visit the current node.
3) Traverse the left subtree of the current node.

Second, after traversing the tree please explain any relationship between this algorithm and preorder, inorder and/or postorder.



6) Assume P points to a node of a threaded binary tree. The tree is assumed to be threaded in inorder. What node does “Q” point to after executing the algorithm? You may assume one of the six tree traversal algorithm utilized in class.

Algorithm Mystery 2:

```
Q := P;
if P.RTag = “+” then
        Q := P.RLink;
else
        while Q.LTag <> “+” loop Q := Q.LLink; end loop;
        Q := Q.LLink;
end if;
```

Section 2: Answer any two of questions 5 – 8.

7) Write an algorithm to insert a new node pointed to by Pt as the left subtree of the node pointed to by Ip in a threaded tree. The tree should be threaded with the inorder predecessor and successor as developed in class. The new node may be either a leaf or an interior node in the resulting tree. The node format follows:

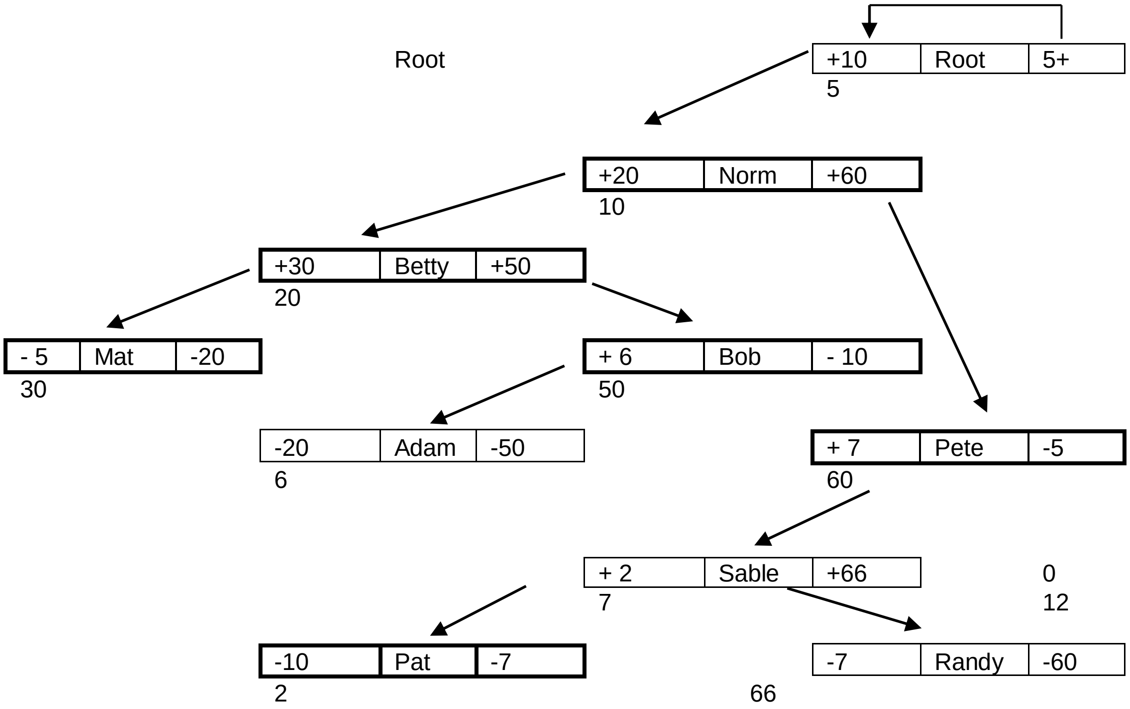
LTag	LLink	Key	Other key data	RLink	RTag

8) **Write a detailed recursive algorithm or procedure in Ada, Java, "C" or pseudo code to determine if a desired key appears in a binary search tree in the most efficient fashion**. If found, print the key data. If not found, print an appropriate message. Assume the tree is not threaded and has the following note format.

LLink	Key	Other key data	RLink

9) Traverse the tree of question 7 first in preorder, followed by inorder, then post order.

10) Assume the following tree is threaded with the inorder predecessor and inorder successor as developed in class (the treads are not drawn). **Write an iterative algorithm to find the inorder successor taking advantage of the threads**.



*** Shows primary links but only address for threads.

11) First exhibit the tree of question 7 after deleting Sable using the algorithm developed to delete from a threaded tree in class. This algorithm is usually associated with search trees. This is not an alphabetic search tree. Second, utilizing the tree completed for part "A" as a starting point, exhibit the results a second time after deleting Betty using the algorithm developed to delete from a threaded tree in class.

12) Assume a hash table of size eight using the random probe technique to handle collisions. Calculate the first two random numbers generated by our algorithm if the seed is 5 (not 1).

13) Compare and contrast the expected performance of the linear and random probe techniques. Your discussion should include the effect caused by the size of the table and the load factor. It should also discuss the anticipated characteristics with respect to clustering as the table fills for both the linear and random probe techniques.

Section 3: Answer one of questions 8 – 10.

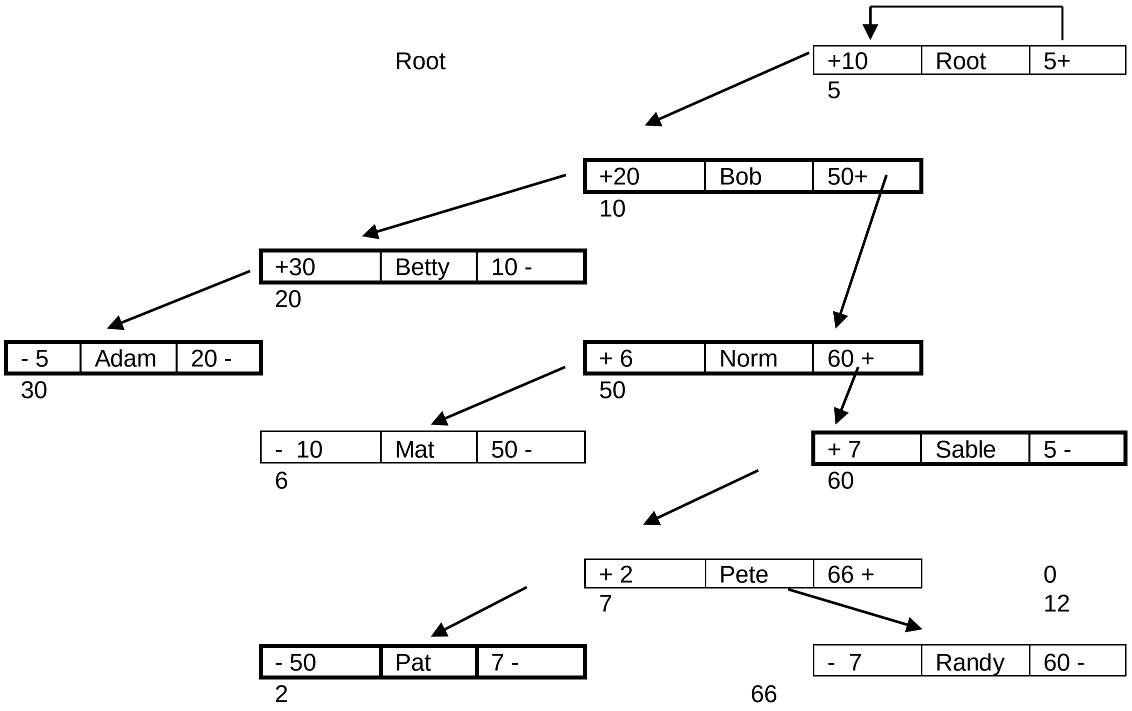
5) Write a detailed iterative or recursive algorithm to do a binary search on a sequentially allocated table by adjusting an upper and lower bound each time a portion of the table is eliminated.

6) Proper coding techniques may substantially reduce the time it takes to do a binary search on a sequentially allocated table. Explain in detail as many of these techniques covered in class as possible explicitly stating why and by how much (quantitatively) each technique reduces the time required to complete the binary search.

7) A binary search is not always the fastest way to search a sequentially allocated main memory table. First consider tables whose size is less than about 50 items. Explain why a sequential search may be faster if each element in the table has an equal probability of being the desired item. Second explain a method for searching a sequentially allocated table which may be faster than a binary search when the data obeys **W. P. Hesing's "80/20**" rule (or a Pareto distribution). How might tables fitting the "80/20" rule adjust their selves over time if the reference frequency for table contents is subject to change?

Section 2: Answer two of questions 5 – 7.

8) Assume the following tree is threaded with the inorder predecessor and inorder successor as developed in class (the treads are not drawn). Write an iterative algorithm to find the inorder predecessor taking advantage of the threads.



*** Shows primary links but only address for threads.

9) Exhibit the data structure and trees produced by Algorithm Equivalence for the following relations given jobs A, B, C, D, E, F, G, H, J, K, L and M after processing all equivalence relations in the sequence they appear on the test from left to right.

D = G, A = B, G = H, D = F, K = L, F = E, H = J, A = K, K = M.

Parent(K)												
K	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**J**	**K**	**L**	**M**

10) Create a B-Tree of order 6 using the following data in the indicated order left to right. A, B, G, K, D, H, M, L S, R, X, C, E, N, and P. You must show complete trees for all intermediate steps. Start/draw a complete new tree each time a node splits.

Section 3: Answer one of questions 8 – 10.

11) Write a detailed iterative or recursive algorithm to do a binary search on a sequentially allocated table by adjusting an upper and lower bound each time a portion of the table is eliminated.

12) Proper coding techniques may substantially reduce the time it takes to do a binary search on a sequentially allocated table. Explain in detail as many of these techniques covered in class as possible explicitly stating why and by how much (quantitatively) each technique reduces the time required to complete the binary search.

13) A binary search is not always the fastest way to search a sequentially allocated main memory table. First consider tables whose size is less than about 50 items. Explain why a sequential search may be faster if each element in the table has an equal probability of being the desired item. Second explain a method for searching a sequentially allocated table which may be faster than a binary search when the data obeys **W. P. Hesing's "80/20"** rule (or a Pareto distribution). How might tables fitting the "80/20" rule adjust their selves over time if the reference frequency for table contents is subject to change?

14) Assume a hash table of size eight using the linear probe technique to handle collisions. The contents of the hash table were initialized to spaces prior to inserting and deleting records. Utilize "&" appropriately in completing your answer.

A) Exhibit the contents of the table on the answer sheet as Table I after performing all of the following operations. Insert Tom at 3, Mary at 4, Bob at 6, Sue at 3, Sam at 8 and Jill at 8. Delete Sue.

B) Insert Anna at 2 and delete Bob from the table from part A. Exhibit the results as Table II on the answer sheet.

Partial answers:

1	Jill
2	
3	Tom
4	Mary
5	&
6	Bob
7	
8	Sam

Table I

1	Jill
2	Anna
3	Tom
4	Mary
5	
6	
7	
8	Sam

Table II

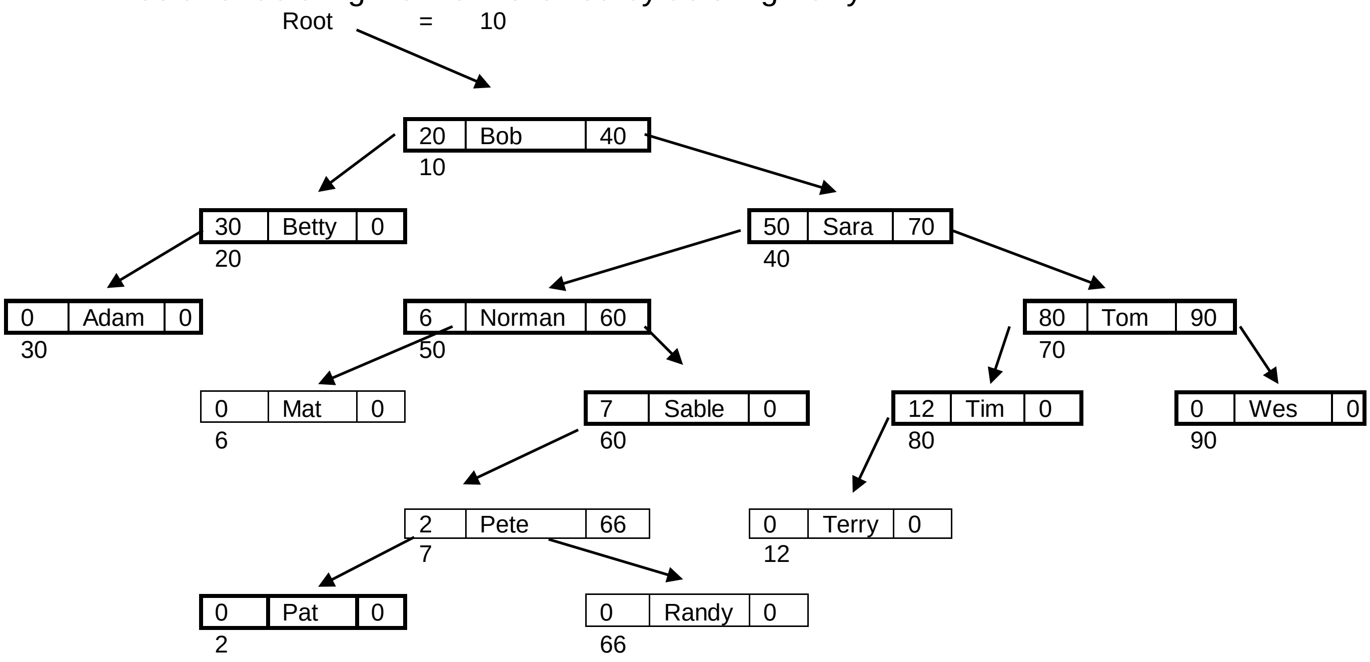
Answer any 5 of the following 7 questions. ***A professional should be able to answer every question***! Number your questions on the answer sheets in ascending numeric order from 1 through 7 inclusive. Clearly write “Delete” on the answer sheets by the 2 questions you do not wish graded. Leave at least a one-inch margin at the top of every page. Start each question you answer on a new page or very neatly on the same page with the preceding question. You may delete multiple questions on the same page. Do not write on the back of pages. Staple your answer sheets (in ascending numeric order) on top of the test in the upper left hand corner. Write your name in the upper right hand corner of the answer sheets (first page). Turn the stapled bundle over and write your name in the upper right hand corner on the back of the test. A five point Road Map Fee (RMF) will be deducted for each instance of failure to follow instructions. You will not receive credit for material I cannot read or that is obstructed from my view. **Please note some questions require substantially more time to answer than others!**

1) Discuss the division remainder method for calculating hash addresses in detail. Your discussion should include but not be limited all restrictions associated with the table size, selecting the bits for the key, limitations imposed by different data types (e.g., character, numeric, etcetera), and utilizing keys whose size exceeds the number of bits in an integer. Include whole record insertion, deletion, reuse of deleted record space, and determining a record does not exist. Assume no collisions.

2) Our client, a text book publisher, needs to develop an online testing program producing a different test consisting of 100 random questions for each participant. It has been suggested the creation of a static test bank (questions never change) of 5,000 questions with answers (true/false, multiple choice, and matching). A different seed will be used in a random number generator for each student taking the test to initialize the random number generator. Each test will consist of questions from the test bank corresponding to 100 random numbers generated by our random number generator. The test will be graded electronically. When each test is shipped, the seed used to initialize the random number generator will be included as part of the test number. That way, given the test number, the answers can be looked up dynamically in the test bank and we will not have to provide answer sheets. In addition, as long as the test bank is not modified (static) we can generate the same test for reference in the future as long as we know the seed. **State all restrictions on the random number generator and why they exist. Next suggest an algorithm to create the test and any restrictions required to make sure no test question is selected from the test bank more than once for a given student.** Hint: we covered a suitable algorithm in class.

3) Write an algorithm to insert a new node in a pointed to by Pt as the left subtree of the node pointed to by Ip. The new node may be either a leaf or an interior node of a tree threaded with the inorder predecessor and successor as developed in class.

4) Assume the following binary search tree. Exhibit (draw) the tree after deleting Norman using the algorithm developed in class (no modifications are allowed). Now exhibit the tree after deleting Norman followed by deleting Terry.



5) Traverse the tree in question 4 printing the contents of the nodes in inorder then post order.

14) Choice (do only one of the following):

A) Write an iterative algorithm using a stack to traverse a non-threaded binary tree in applying the following steps at each node: a) traverse the right subtree, b) traverse the left subtree, and c) visit the node (RLV). You may use the notation T => S to indicate pushing the value T into stack S or the notation (T, V) => S to indicated pushing the values T and V into the stack S. The notation T <= S and (T,V) <= S indicates popping the stack to recover one or two values. Assume the tree is pointed to by root.

B) Assume a binary tree threaded in inorder and you are currently at the node pointed to by P. Write an algorithm to find the preorder successor (P*) utilizing the threads efficiently. Assume the head of the tree is at location root.

15) Exhibit the data structures and trees produced by Algorithm Equivalence for the following relations given jobs A, B, C, D, E, F, G, H, J and K.

A = B, F = G, D = E, C = D, E = A, F = J, and D = K

Parent(K)										
K	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**J**	**K**

16) Compare and contrast insertion, deletion, searching, and the ability to generate a sorted list of the contents of a binary search tree of N items with a sequentially allocated array (linear list) maintained in random order. You must use quantitative measures to receive credit. A include a table summarizing your results as part of your answer.

	Minimum Probes Search Find	Maximum Probes Search Find	Average Probes Search Find	Expected Insertion Overhead	Expected Deletion overhead	Minimum Insertion Overhead	Maximum Insertion Overhead	Expected Insertion Overhead
Search Tree								
Random Order List								

A) Choice (do one of the following):

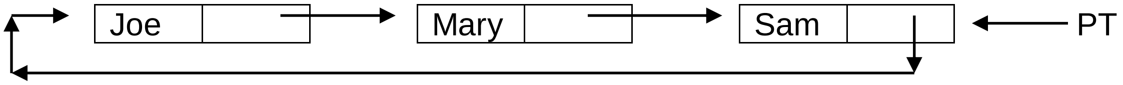
a) Discuss the division remainder technique for computing hash addresses in detail. Be sure to mention all advantages and disadvantages.

b) Explain in detail how the random probe technique is used in collision handling. Cover looking up existing records, records that do not exist, whole record insertion, and whole record deletion.

c) List and discuss the restrictions placed on a random number generator for the random probe technique. Explain how each restriction is met using the random number generator developed in class. How are the actual hash address calculated during for successive collisions?

d)

B) Write an algorithm to insert a new node in a circular list in lexicographic (ascending) order. Remember there are four cases. A sample circular list follows.



```
Hint:
P <= avail
p.info ← y
if pt = Ω   -- insert in empty list
    p.next ← pt ← p
else
    if p.info > pt.info
        --Inset on rear/right (finish the algorithm), new last item
    Else
        Pt1 ← pt.next
        If P.info < pt1.info
            --Inset in front/left (finish the algorithm), new first item
        Else
            --Must insert as an interior node after locating its position
            --(finish the algorithm)
        End if
    End if
End if
```

17) Choice (do only one of the following):

C) Write an iterative algorithm to traverse a non-threaded binary tree in the reverse of preorder: RLV. You may use the notation T => S to indicate pushing the value T into stack S or the notation (T, V) => S to indicated pushing the values T and V into the stack S. The notation T <= S and (T,V) <= S indicates popping the stack to recover one or two values. Assume the tree is pointed to by root.

D) Assume a binary tree threaded in inorder and you are currently at the node pointed to by P. Write an algorithm to find the in order predecessor ($P) utilizing the threads efficiently. Assume the head of the tree is at location root.

18) Write an algorithm to determine if two threaded binary trees pointed to by R1 and R2 are equivalent.

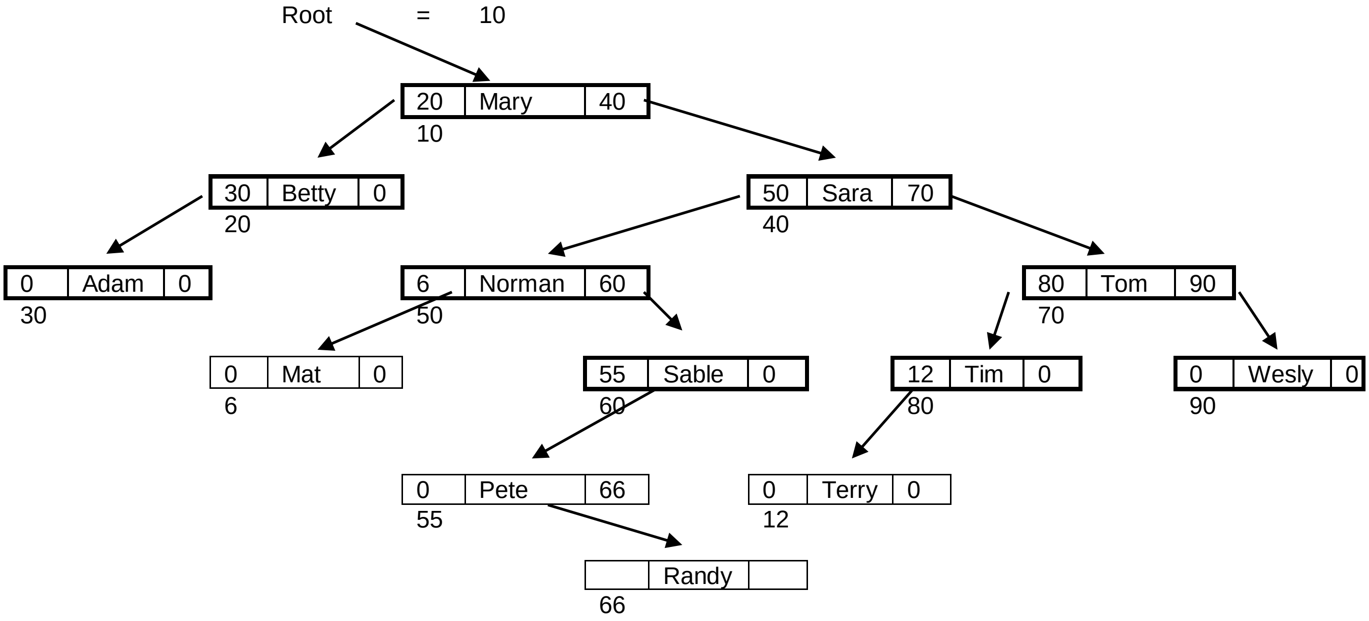
19) Write a **detailed recursive** binary search algorithm to locate a specific person in a threaded binary (alphabetic) search tree. Assume each node contains a left link, right link, left tag, right tag, name field, and phone number. The tree head is located at location root and the tree is threaded in inorder. Allow for the case the person is not present. Your algorithm must be explicit stating exactly what to do in terms of examining the key (name) field and link or tag fields.

20) Write an algorithm to traverse a non-threaded binary tree using a stack efficiently if the traversal is defined recursively for each node as: traverse the right subtree, visit the node, traverse the left subtree. This algorithm is effectively the reverse of the inorder algorithm developed in class. You may use the notation pt => stack and pt <= stack to indicate pushing and popping pt into and from the stack respectively.

21) Write an algorithm to attach a node pointed to by Q as the right subtree of the node pointed to by P in a threaded binary tree. Note the node being inserted may be a leaf or an internal node. The tree is assumed to be threaded in inorder with an appropriate tree head node at location root. The node format is:

LTag	LLink	Info	RLink	RTag

22) Assume the following binary search tree. Exhibit (draw) the tree after deleting Sara. Now exhibit the tree after deleting Mary according to the binary search tree deletion algorithm taught in class.



23) Choice (do only one of the following):

24) Discuss the linear probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

25) Discuss the random probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

26) Discuss the logic used to develop the random number generator for the random probe technique. You must include any restrictions on the random number generator, table size, etcetera. I am not looking for a regurgitation of the algorithm. The assumption is you know the algorithm and are now explaining the logic for the algorithm

27) Choice (do only one of the following):

28) Explain in great detail the characteristics required of a random number generator to support the random probe method for collisions. You discussion should include how many times a random number can be generated and the actual calculation of all hash addresses after the initial hash address resulting in the collision. You might desire to include the algorithm for generating random number for use with the technique but that would not be sufficient to successfully address the question. How can we tell when the hash table is full and when a key is not in the hash table?

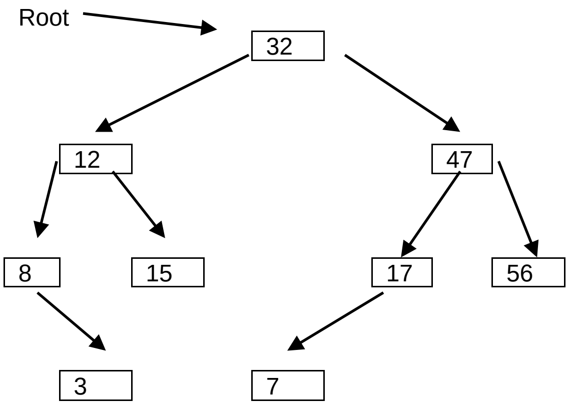
29) Explain in detail how to insert, delete, and locate keys using the random probe technique. Your discussion must include all required information with respect to the hash table when a key is deleted. How can we tell when the hash table is full and when a key is not in the hash table?

30) Write a **detailed recursive** algorithm to locate a specific person in a threaded binary (alphabetic) search tree. Assume each node contains a left link, right link, left tag, right tag, name field, and phone number. The tree head is located at location root. Allow for the case the person is not present. Your algorithm must bed explicit stating exactly what to do in terms of examining the key (name) field and link or tag fields.

31) Choice (do only one of the following):

A) Write an algorithm to determine if the threaded binary trees pointed to by Root1 and Root2 are equivalent.

B) List the contents of the following tree in inorder, preorder, and postorder.



e) Given “A: array (Sam..Ken, 5..19, -18..9, 0..14, -6..-4, -8..0) of integer” write a formula evaluating all constants to locate A[B, C, D, E, F, G] assuming lexicographic storage order and that each integer occupies 4 bytes of memory (32 bit integers). <u>You must show your work to receive credit</u>. <u>*No credit will be allocated for simply supplying an answer.*</u> The programmer defined enumeration type is defined as: type Name is (Tom, Betty, Sam, Mary, Happy, Sally, Bob, Jill, Tim, Barbie, Ken, Gary). Assume the data type "Name" is mapped by the translator into integers in the range 0..11. You do not actually have to do the multiplications so long as you have the formulas written correctly and all constants have been expanded except for the final multiplications.

f) Choice (do only one of the following):

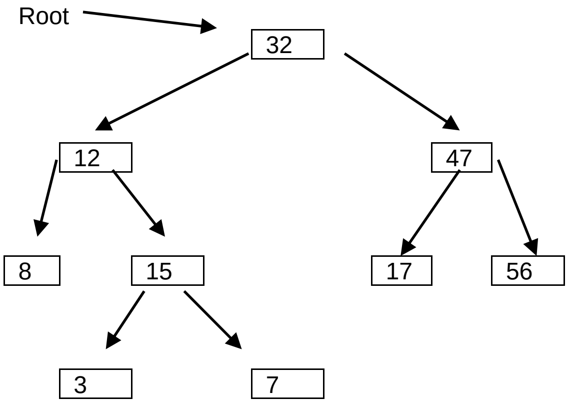
32) Discuss the linear probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

33) Discuss the random probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

34) Discuss the logic used to develop the random number generator for the random probe technique. You must include any restrictions on the random number generator, table size, etceteras. I am not looking for a regurgitation of the algorithm. The assumption is you know the algorithm and are now explaining the logic of each step.

3) Write an algorithm to traverse a non-threaded binary tree using a stack efficiently if the traversal is defined recursively for each node as: traverse the right subtree, visit the node, traverse the left subtree. This algorithm is effectively the reverse of the inorder algorithm developed in class. You may use the notation pt => stack and pt <= stack to indicate pushing and popping pt into and from the stack respectively.

4) Print the contents of the following tree in inorder, preorder, and postorder.



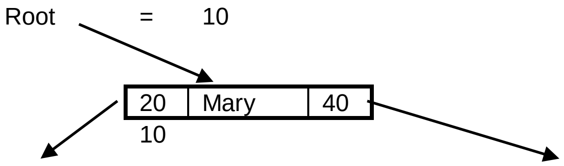
5) Write an algorithm to insert a new node pointed to by Q as the right subtree of the node pointed to by P in a binary tree threaded in inorder. The inserted node may be a leaf or interior node.

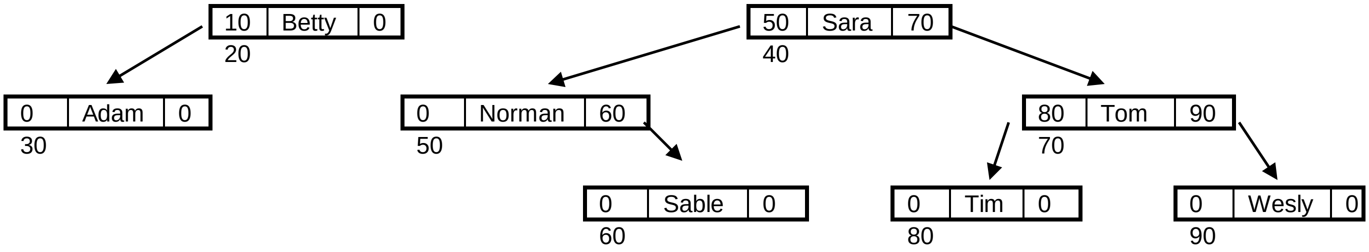
6) Write an algorithm to find the inorder successor of the node at location P in a threaded binary tree. You must allow the search to begin at any (random) node in the tree taking advantage of the threads. The tree was created using inorder predecessor and inorder successor threads for null left and right links.

7) Write a **recursive** algorithm to locate a specific person in a binary (alphabetic) search tree. Assume each node contains a left link, right link, name field, and phone number. The tree is pointed to by root. Allow for the case the person is not present.

    1) Discuss “B” trees in detail including their motivations, how to build them, search them, and their efficiency with respect to use with moving arm disks.

8) Using the algorithm developed in class for binary search tree deletion, exhibit the following tree after Betty and Sara have been deleted from the alphabetic (binary) search tree.

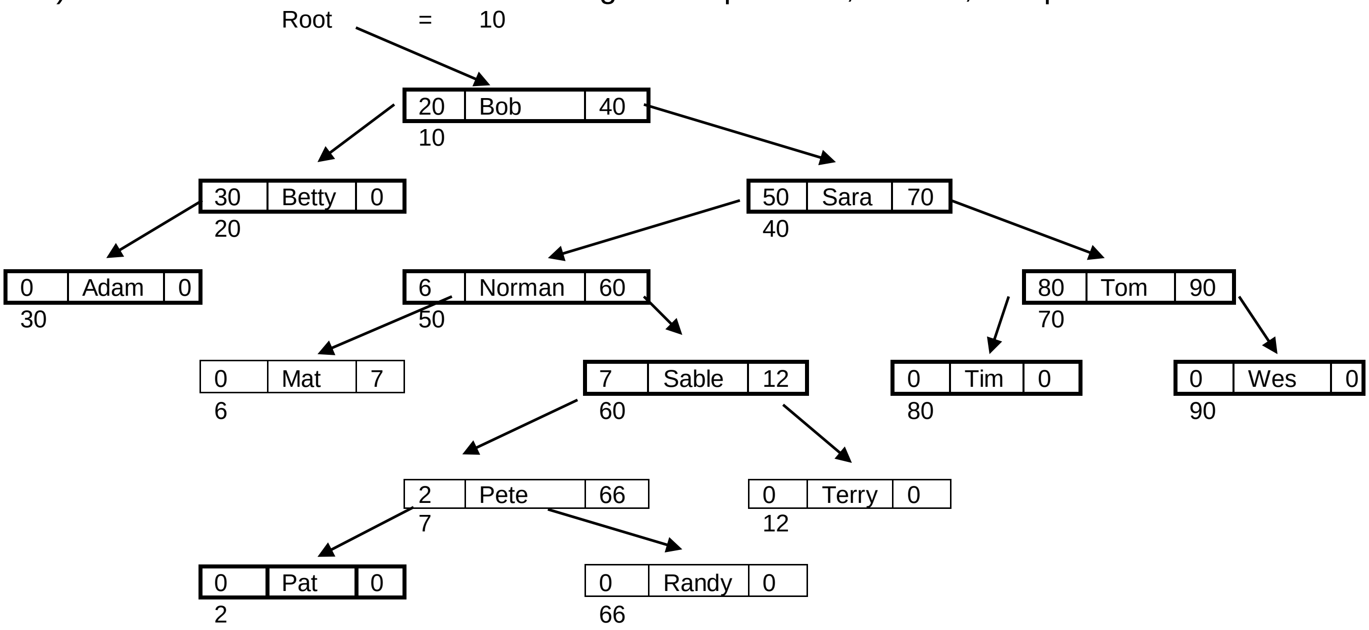




9) Evaluate the space required for the stack when traversing a non-threaded binary search tree in pre-order. *Be sure your discussion clearly states the expected, minimum, and maximum size for the stack and the characteristics of the tree resulting in these stack sizes.* The stack may be explicit (you created it) or implicit (as a result of using recursive subprogram invocations in a recursive programming language).

I have made the first pass to create test 3 in Data Structures. At present the test stresses pages 74 – 92 and pages 136-169. I anticipate we will complete pages 93 – 102 prior to the exam.

35) Exhibit the contents of the following tree in preorder, inorder, and postorder.



9) Choice (do only one of the following):

A) Develop a formula to calculate addresses to store/retrieve entries in the following lower triangular matrix starting at memory address a_0 in column major order, i.e., write a formula to find Loc(A[j,k]) = ---? Column major order will store the first column in memory, followed by the second column starting at the base address etceteras till all columns are stored { i.e., starting at the base address, the first items stored will be A(0,0), A(1,0), through A(n,0) followed by A(0,1) through A(n-1,1, etceteras}. Note we only need to store the non zero entries A(j,k) for 0 <= j <= k <= n. Do not store the lower (all zero) portion of the matrix.

A(0,0)	A(0,1)	A(0,2)	A(0,3)	°°°	A(0,2)
A(1,0)	A(1,1)	A(1,2)	A(1,3)	°°°	0
°°°					
A(n-2,0)	A(n-2,01)	A(n-2,2)	0	°°°	0
A(n-1,0)	A(n-1,1)	0	0	°°°	0
A(n,0)	0	0	0	°°°	0

Assume each entry in the array consumes "C" units of storage.

Hint: $$\text{LOC}(A[J,K]) = \text{base} + \left(\sum_{M=0}^{K} (4-M) + J\right) * C$$

B) Assume a 1000 by 1000 logical array representing weather stations. At any point in time only 800 stations will be queried for information to drive our weather simulation. We have insufficient compute power to utilize more measurements (even though more measurements are desirable). Develop a data structure to minimize the actual amount of storage required to represent the array. Explain your storage scheme using a diagram. State specifically how much storage is required and compute the savings relative to reserving storage for a full array. Assume an integer/pointer is 4 bytes in length and the storage to represent a weather station is 650 bytes. The full array would require 1000 x 1000 x 650 bytes.

Hint: 650,000,000 versus (1000+1000)*4 + 800*(650+8)=526,400

C) Given "A: array (5..9, -8..9, 0..4, -6..-4, red..green) of integer" write a formula evaluating all constants to locate A[J, K, L, M, N] assuming lexicographic storage order and that each integer occupies 4 units of memory. You must show your work to receive credit. The programmer defined enumeration type is defined as: type color is (blue, red, yellow, violet, black, green, white). You do not actually have to do the multiplications so long as you have the formulas written correctly and all constants have been expanded except for the final multiplications.

Hint: LOC(A[J,K,L,M,N]) = BASE + 1350(J-5) + 75(K+8) + 15l + 5(M+6) + (N-1)

2) Choice (do one of the following):

3) Discuss the division remainder technique for computing hash addresses in detail. Be sure to mention all advantages and disadvantages.

4) Explain in detail how the random probe technique is used in collision handling. Cover looking up existing records, records that do not exist, whole record insertion, and whole record deletion.

5) List and discuss the restrictions placed on a random number generator for the random probe technique. Explain how each restriction is met using the random number generator developed in class. How are the actual hash address calculated during for successive collisions?

Write an algorithm to traverse a non-threaded binary tree using a stack efficiently if the traversal is defined recursively for each node as traverse the right subtree, visit the node, traverse the left subtree. This algorithm is effectively the reverse of the inorder algorithm developed in class. You may use the notation pt => stack and pt <= stack to indicate pushing and popping pt into and from the stack respectively

g) Choice (do only one of the following):

6) Explain in detail how the random probe technique is used in collision handling. Cover looking up existing records, records that do not exist, whole record insertion, whole record deletion, and reducing the search path for existing records after record deletion.

7) Discuss the division remainder technique for computing hash addresses in detail. Be sure to mention all advantages and disadvantages.

8) Write an algorithm to determine if two threaded binary trees are equivalent.

8) Choice (do only one of the following):

E) Write an algorithm to traverse a non-threaded binary tree in the reverse of inorder: RVL. <u>Your algorithm must explicitly perform all stack operations</u>. You may not use the simplified notation in class A <= y or y <= A to indicate pushing or popping variable y from or into stack A. Assume the tree is pointed to by root.

F) Assume a threaded binary tree. Write an algorithm to find the next node in preorder utilizing the threads efficiently. Assume the tree is pointed to by root and you are at location Q.

9) Choice (do only one of the following):

G) Write an iterative algorithm to traverse a non-threaded binary tree in the reverse of preorder: . Your algorithm must explicitly perform all stack operations. You may not use the simplified notation in class A <= y or y <= A to indicate pushing or popping variable y from or into stack A. Assume the tree is pointed to by root.

H) Assume a threaded binary tree. Write an algorithm to find the next node in preorder utilizing the threads efficiently. Assume the tree is pointed to by root and you are at location Q.

10) Choice (do only one of the following):
A) Discuss the square and extract N bits hash method in detail. Be sure to cover all issues with respect to numeric and alphabetic keys to minimize collisions.

B) Discuss the Linear Probe technique in detail with respect to insertion, deletion, and finding keys. Your discussion should include reclaim and using deleted space. Explain why or why not the time to find keys entered into the table as a result of a collision can be shortened if a key in the collision path is deleted.

11) Given “A: array (Sam..Ken, 5..19, -18..9, 0..14, -6..-4, -8..0) of integer” write a formula evaluating all constants to locate A[B, C, D, E, F, G] assuming lexicographic storage order and that each integer occupies 4 bytes of memory (32 bit integers). You must show your work to receive credit. ***No credit will be allocated for simply supplying an answer.*** The programmer defined enumeration type is defined as: type Name is (Tom, Betty, Sam, Mary, Happy, Sally, Bob, Jill, Tim, Barbie, Ken, Gary). Assume the data type "Name" is mapped by the translator into integers in the range 0..11. You do not actually have to do the multiplications so long as you have the formulas written correctly and all constants have been expanded except for the final multiplications.

C) Choice (do only one of the following):

A) Given “A: array (Mary..Ken, 15..19, -18..10, 0..14, -6..-2, -8..0) of integer” write a formula evaluating all constants to locate A[B, C, D, E, F, G] assuming lexicographic storage order and that each integer occupies 4 bytes of memory (32 bit integers). You must show your work to receive credit. ***No credit will be allocated for simply supplying an answer.*** The programmer defined enumeration type is defined as: type Name is (Tom, Betty, Sam, Mary, Happy, Sally, Bob, Jill, Tim, Barbie, Ken, Gary). Assume the data type "Name" is mapped by the translator into integers in the range 0..11. You do not actually have to do the multiplications so long as you have the formulas written correctly and all constants have been expanded except for the final multiplications. You may provide your answer in the form “d2 * d3” or “23 * 15” without actually doing the arithmetic! You must however show the value of all variables like d3.

B) Captain Kirk wants a new 3-dimensional display created for the Bridge on the Star Ship Enterprise. The Enterprise should always be located in the center at coordinates (0, 0, 0). A new Ensign has suggested solving the problem using a 3-dimensional array as follows: display: array (-100,000..100,000; 100,000..100,000; 100,000..100,000) of SpaceType. Our chief science officer Mr. Spock has noted we normally do not have that much available main memory and we really only need to plot about 50,000 entities in the space for both navigation and fire control. 1) Suggest an alternative way to represent the 3-dimensional array using Ada or pseudo code (Wallgol) to allocate the space. 2) You must define the format for representing an item in the space you defined. 3) Finally you must numerically evaluate the space requirements for you solution and compare them to the Ensign's proposal. For full credit, you better save a lot of storage on the main computer.

D) Choice (do only one of the following):

12) Discuss the linear probe technique for handling collisions in detail. Your discussion must include a) insertion, b) how do I know when I have found a spot to insert, c) deletion, e) searches where the key is found, f) searchers where the key is not found, g) reclaiming space, and h) listing the keys in sorted order.

13) Discuss the random probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

14) Explain in great detail the characteristics required of a random number generator to support the random probe method for collisions. You discussion should include how many times a random number can be generated and the actual calculation of all hash addresses after the initial hash address resulting in the collision. You might desire to include the algorithm for generating random number for use with the technique but that would not be sufficient to successfully address the question. How can we tell when the hash table is full and when a key is not in the hash table?

E) Choice (do one of the following):

h) Discuss the division remainder technique for computing hash addresses in detail. Be sure to mention all advantages and disadvantages.

i) Explain in detail how the random probe technique is used in collision handling. Cover looking up existing records, records that do not exist, whole record insertion, and whole record deletion.

j) List and discuss the restrictions placed on a random number generator for the random probe technique. Explain how each restriction is met using the random number generator developed in class. How are the actual hash address calculated during for successive collisions?

15) Captain Kirk wants a new 3-dimensional display created for the Bridge on the Star Ship Enterprise. The Enterprise should always be located in the center at coordinates (0, 0, 0). A new Ensign has suggested solving the problem using a 3-dimensional array as follows: display: array (-100,000..100,000; 100,000..100,000; 100,000..100,000) of SpaceType. Our chief science officer Mr. Spock has noted we normally do not have that much available main memory and we really only need to plot about 50,000 entities in the space for both navigation and fire control. 1) Suggest an alternative way to represent the 3-dimensional array using Ada or pseudo code (Wallgol) to allocate the space. 2) You must define the format for representing an item in the space you defined. 3) Finally you must numerically evaluate the space requirements for you solution and compare them to the Ensign's proposal. For full credit, you better save a lot of storage on the main computer.

16) Choice (do only one of the following):

17) Discuss the linear probe technique for handling collisions in detail. Your discussion must include a) insertion, b) how do I know when I have found a spot to insert, c) deletion, e) searches where the key is found, f) searchers where the key is not found, g) reclaiming space, and h) listing the keys in sorted order.

18) Discuss the random probe technique for handling collisions in detail. Your discussion must include insertion, deletion, searches where the key is found, searchers where the key is not found, reclaiming space, and listing the keys in sorted order.

19) Explain in great detail the characteristics required of a random number generator to support the random probe method for collisions. You discussion should include how many times a random number can be generated and the actual calculation of all hash addresses after the initial hash address resulting in the collision. You might desire to include the algorithm for generating random number for use with the technique but that would not be sufficient to successfully address the question. How can we tell when the hash table is full and when a key is not in the hash table?

20) Captain Kirk wants a new 3-dimensional display created for the Bridge on the Star Ship Enterprise. The Enterprise should always be located in the center at coordinates (0, 0, 0). A new Ensign has suggested solving the problem using a 3-dimensional array as follows: display: array (-100,000..100,000; -100,000..100,000; -100,000..100,000) of SpaceType. Our chief science officer Mr. Spock has noted we normally do not have that much available main memory. He has managed to represent the essential data in a single two dimensional NxN lower triangular matrix as shown below. **Please develop a formula to locate any random element Space[row, col] in the matrix. Specifically your formula must determine Loc(Space[row, col]) for any random row and column**. You must evaluate all constants in your equation explaining/showing your work. Each element of the matrix consumes "C" units of main memory.

Space[0,0]					
Space[1, 0]	Space[1,1]]				
Space[2,0]	Space[2, 1]]	Space[2, 2]]			
ooo					
Space[N-1,0]	Space[N-1, 1]	Space[N-1, 2]	Space[N-1, 3]	ooo	Space[N-1, N-1]

A) Compare and contrast the advantages of threaded versus non-threaded trees. Consider execution speed, storage utilization, amount of code required for implementation, and potential problems in debugging your application.

36) Write a **detailed recursive** binary search algorithm to locate a specific person in a threaded binary (alphabetic) search tree. Assume each node contains a left link, right link, left tag, right tag, name field, and phone number. The tree head is located at location root and the tree is threaded in inorder. Allow for the case the person is not present. Your algorithm must be explicit stating exactly what to do in terms of examining the key (name) field and link or tag fields.

A) Discuss the random probe technique for handling collisions in detail. Your discussion must include but is not limited to a) insertion, b) how do I know when I have found a spot to insert, c) deletion, e) searches where the key is found, f) searchers where the key is not found, g) reclaiming space, and h) listing the keys in sorted order, making sure you do not visit the same place twice. You will need to discuss the desirable properties of the random number generator but do not discuss a method to implement the generator.

B) Write an algorithm to determine if the threaded binary trees pointed to by Root1 and Root2 are equivalent.

C)

37) Write a **detailed recursive** algorithm to locate the record containing the highest (largest key) value of the name field in a threaded binary (alphabetic) search tree. Assume each node contains a left link, right link, left tag, right tag, name field, and phone number. The tree head is located at location root and the tree is threaded in inorder. Allow for the case the tree is empty. Your algorithm must be explicit stating exactly what to do in terms of examining the key (name) field and link or tag fields.