Chapter 9

Data Structures

***1.** Consider the following tree

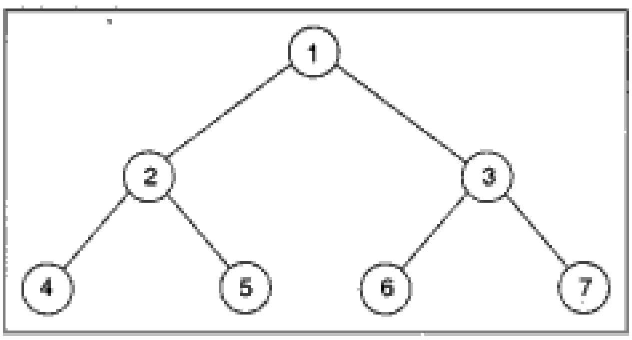


Fig. 9.1

If the post order traversal gives `a b - c d * +` then the label of the nodes `1,2,3,...` will be

(a) `+, -, *, a, b, c, d`
(b) `a, -, b, +, c, *, d`
(c) `a, b, c, d, -, *, +`
(d) `-, a, b, +, *, c, d`

***2.** Consider the following tree.

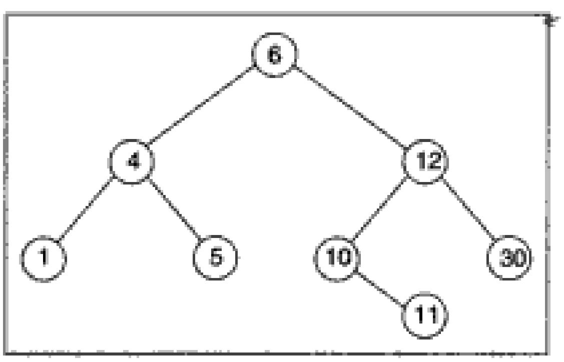


Fig. 9.2

If this tree is used for sorting, then a new number 8 should be placed as the

(a) left child of the node labeled 30 (b) right child of the node labeled 5

(c) right child of the node labeled 30 (d) left child of the node labeled 10

3. The initial configuration of a queue is *a*, *b*, *c*, *d*, ('*a*' is in the front end). To get the configuration *d*, *c*, *b*, *a*, one needs a minimum of

(a) 2 deletions and 3 additions (b) 3 deletions and 2 additions

(c) 3 deletions and 3 additions (d) 3 deletions and 4 additions

***4.** The number of possible ordered trees with 3 nodes A, B, C is

(a) 16 (b) 12 (c) 6 (d) 10

5. The number of swappings needed to sort the numbers 8, 22, 7, 9, 31, 19, 5, 13 in ascending order, using bubble sort is

(a) 11 (b) 12 (c) 13 (d) 14

***6.** Given two sorted list of size 'm' and 'n' respectively. The number of comparisons needed in the worst case by the merge sort algorithm will be

(a) m × n (b) maximum of m, n

(c) minimum of m, n (d) m + n – 1

7. If the sequence of operations - push(1), push(2), pop, push(1), push(2), pop, pop, pop, push(2), pop, are performed on a stack, the sequence of popped out values are

(a) 2, 2, 1, 1, 2 (b) 2, 2, 1, 2, 2

(c) 2, 1, 2, 2, 1 (d) 2, 1, 2, 2, 2

***8.** A hash table with 10 buckets with one slot per bucket is depicted in Fig. 9.3. The symbols, S1 to S7 are initially entered using a hashing function with linear probing. The maximum number of comparisons needed in searching an item that is not present is

(a) 4 (b) 5 (c) 6 (d) 3

0	S7
1	S1
2	
3	S4
4	S2
5	
6	S5
7	
8	S6
9	S3

Fig. 9.3

***9.** A binary tree in which every non-leaf node has non-empty left and right subtrees is called a strictly binary tree. Such a tree with 10 leaves

(a) cannot have more than 19 nodes (b) has exactly 19 nodes

(c) has exactly 17 nodes (d) cannot have more than 17 nodes

***10.** The depth of a complete binary tree with 'n' nodes is (log is to the base two)

(a) log(n+1) – 1 (b) log(n)

(c) log(n–1) + 1 (d) log(n) + 1

11. Preorder is same as

(a) depth-first order (b) breadth-first order

(c) topological order (d) linear order

***12.** Which of the following traversal techniques lists the nodes of a binary search tree in ascending order?

(a) Post-order (b) In-order (c) Pre-order (d) None of the above

***13.** The average successful search time taken by binary search on a sorted array of 10 items is

(a) 2.6 (b) 2.7 (c) 2.8 (d) 2.9

***14.** A hash function `f` defined as `f(key) = key mod 7`, with linear probing, is used to insert the keys `37, 38, 72, 48, 98, 11, 56`, into a table indexed from 0 to 6. What will be the location of key 11?

(a) 3 (b) 4 (c) 5 (d) 6

***15.** The average successful search time for sequential search on 'n' items is

(a) n / 2 (b) (n–1) / 2 (c) (n+1) / 2 (d) log (n) + 1

16. The running time of an algorithm T(n), where 'n' is the input size is given by

$$T(n) = 8T(n/2) + qn, \text{ if } n > 1$$
$$p, \text{ if } n = 1$$

where `p,q` are constants. The order of this algorithm is

(a) n^2 (b) n^n (c) n^3 (d) n

***17.** Let m, n be positive integers. Define Q(m,n) as

$$Q(m, n) = 0, \text{ if } m < n$$
$$Q(m - n, n) + p, \text{ if } m \geq n$$

Then Q(m, 3) is (a div b, gives the quotient when a is divided by b)

(a) a constant (b) p × (m mod 3) (c) p × (m div 3) (d) 3 × p

***18.** Six files `F1,F2,F3,F4,F5` and `F6` have 100, 200, 50, 80, 120, 150 number of records respectively. In what order should they be stored so as to optimize access time? Assume each file is accessed with the same frequency.

(a) `F3, F4, F1, F5, F6, F2`

(b) `F2, F6, F5, F1, F4, F3`

(c) `F1, F2, F3, F4, F5, F6`

(d) Ordering is immaterial as all files are accessed with the same frequency.

***19.** In Qn. 18, the average access time will be

(a) 268 units (b) 256 units (c) 293 units (d) 210 units

***20.** An algorithm is made up of 2 modules `M1` and `M2`. If order of `M1` is `f(n)` and `M2` is `g(n)` then the order of the algorithm is

(a) `max(f(n),g(n))` (b) `min(f(n),g(n))`

(c) `f(n) + g(n)` (d) `f(n) × g(n)`

21. The concept of order (Big O) is important because

(a) it can be used to decide the best algorithm that solves a given problem

(b) it determines the maximum size of a problem that can be solved in a given system, in a given amount of time

(c) it is the lower bound of the growth rate of the algorithm

(d) none of the above

***22.** The running time T(n), where 'n' is the input size of a recursive algorithm is given as follows.

$$T(n) = c + T(n-1), \text{ if } n > 1$$
$$d, \text{ if } n \le 1$$

The order of this algorithm is

(a) n^2 (b) n (c) n^3 (d) n^n

23. There are 4 different algorithms A1, A2, A3, A4 to solve a given problem with the order log(n), log(log(n)), nlog(n), n/log(n) respectively. Which is the best algorithm?

(a) A1 (b) A2 (c) A4 (d) A3

***24.** The number of possible binary trees with 3 nodes is

(a) 12 (b) 13 (c) 5 (d) 15

***25.** The number of possible binary trees with 4 nodes is

(a) 12 (b) 13 (c) 14 (d) 15

26. The time complexity of an algorithm T(n), where n is the input size is given by

$$T(n) = T(n-1) + 1/n, \text{ if } n > 1$$
$$1, \text{ otherwise}$$

The order of this algorithm is

(a) log n (b) n (c) n^2 (d) n^n

27. Sorting is useful for

(a) report generation (b) minimizing the storage needed
(c) making searching easier and efficient (c) responding to queries easily

28. Choose the correct statements.

(a) Internal sorting is used if the number of items to be sorted is very large.
(b) External sorting is used if the number of items to be sorted is very large.
(c) External sorting needs auxiliary storage.
(d) Internal sorting needs auxiliary storage.

29. A sorting technique that guarantees, that records with the same primary key occurs in the same order in the sorted list as in the original unsorted list is said to be

(a) stable (b) consistent (c) external (d) linear

***30.** A text is made up of the characters a, b, c, d, e each occurring with the probability .12, .4, .15, .08 and .25 respectively. The optimal coding technique will have the average length of

(a) 2.15 (b) 3.01 (c) 2.3 (d) 1.78

31. In the previous question, which of the following characters will have codes of length 3?

(a) Only c (b) Only b (c) b and c (d) Only d

***32.** The running time of an algorithm is given by

$$T(n) = T(n-1) + T(n-2) - T(n-3), \text{ if } n > 3$$
$$n, \text{ otherwise.}$$

The order of this algorithm is

(a) n (b) log n (c) n^n (d) n^2

***33.** What should be the relation between T(1), T(2) and T(3), so that Qn. 32, gives an algorithm whose order is constant?

(a) T(1) = T(2) = T(3) (b) T(1) + T(3) = 2T(2)
(c) T(1) – T(3) = T(2) (d) T(1) + T(2) = T(3)

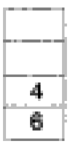
34. The Ackermann's function

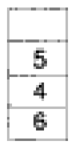
(a) has quadratic time complexity (b) has exponential time complexity
(c) can't be solved iteratively (d) has logarithmic time complexity

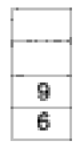
***35.** The order of an algorithm that finds whether a given Boolean function of 'n' variables, produces a 1 is

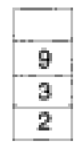
(a) constant (b) linear (c) logarithmic (d) exponential

***36.** In evaluating the arithmetic expression 2 * 3 – (4 + 5), using stacks to evaluate its equivalent post-fix form, which of the following stack configuration is not possible?

(a) 

(b) 

(c) 

(d) 

37. The way a card game player arranges his cards as he picks them up one by one, is an example of

(a) bubble sort (b) selection sort (c) insertion sort (d) merge sort

38. You want to check whether a given set of items is sorted. Which of the following sorting methods will be the most efficient if it is already in sorted order?

(a) Bubble sort (b) Selection sort (c) Insertion sort (d) Merge sort

***39.** The average number of comparisons performed by the merge sort algorithm, in merging two sorted lists of length 2 is

(a) 8/3 (b) 8/5 (c) 11/7 (d) 11/6

40. Which of the following sorting methods will be the best if number of swappings done, is the only measure of efficiency?

(a) Bubble sort (b) Selection sort (c) Insertion sort (d) Quick sort

41. You are asked to sort 15 randomly generated numbers. You should prefer

(a) bubble sort (b) quick sort (c) merge sort (d) heap sort

42. As part of the maintenance work, you are entrusted with the work of rearranging the library books in a shelf in proper order, at the end of each day. The ideal choice will be

(a) bubble sort (b) insertion sort (c) selection sort (d) heap sort

***43.** The maximum number of comparisons needed to sort 7 items using radix sort is (assume each item is a 4 digit decimal number)

(a) 280 (b) 40 (c) 47 (d) 38

44. Which of the following algorithms exhibits the unnatural behavior that, minimum number of comparisons are needed if the list to be sorted is in the reverse order and maximum number of comparisons are needed if they are already in sorted order?

(a) Heap sort (b) Radix sort
(c) Binary insertion sort (d) There can't be any such sorting method

45. Which of the following sorting algorithm has the worst time complexity of `nlog(n)`?

(a) Heap sort (b) Quick sort (c) Insertion sort (d) Selection sort

46. Which of the following sorting methods sorts a given set of items that is already in sorted order or in reverse sorted order with equal speed?

(a) Heap sort (b) Quick sort (c) Insertion sort (d) Selection sort

***47.** Which of the following algorithms solves the all-pair shortest path problem?

(a) Dijkstra's algorithm (b) Floyd's algorithm
(c) Prim's algorithm (d) Warshall's algorithm

***48.** Consider the graph in Fig. 9.4.
The third row in the transitive closure of the above graph is

(a) 1,1,1 (b) 1,1,0
(c) 1,0,0 (d) 0,1,1

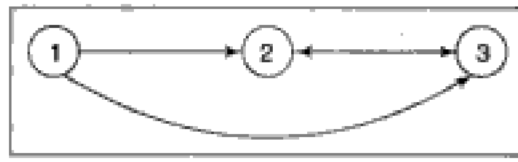


Fig. 9.4

***49.** The eccentricity of node labeled 5 in the graph in Fig. 9.5 is

(a) 6 (b) 7
(c) 8 (d) 5

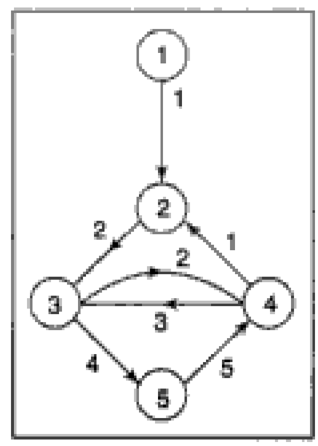


Fig. 9.5

***50.** The center of the graph in Qn. 49 is the node labeled

(a) 1 (b) 2
(c) 3 (d) 4

51. Stack A has the entries `a, b, c` (with a on top). Stack B is empty. An entry popped out of stack A can be printed immediately or pushed to stack B. An entry popped out of stack B can only be printed. In this arrangement, which of the following permutations of `a, b, c` is not possible?

(a) `b a c` (b) `b c a`
(c) `c a b` (d) `a b c`

***52.** In the previous problem, if the stack A has 4 entries, then the number of possible permutations will be

(a) 24 (b) 12 (c) 21 (d) 14

53. The information about an array that is used in a program will be stored in

(a) symbol table (b) activation record (c) system table (d) dope vector

54. Which of the following expressions accesses the $(i,j)^{th}$ entry of a $(m \times n)$ matrix stored in column major form?

(a) `n x (i-1) + j` (b) `m x (j-1) + i`
(c) `m x (n-j) + j` (d) `n x (m-i) + j`

55. Sparse matrices have

(a) many zero entries (b) many non-zero entries
(c) higher dimension (d) none of the above

81. The postfix expression for the infix expression

`A + B* (C+D) / F + D*E` is:

(a) `AB + CD + *F / D + E*`

(b) `ABCD + *F / + DE* +`

(c) `A*B + CD / F*DE ++`

(d) `A + *BCD / F*DE ++`

***82.** Which of the following statements is true?

I. As the number of entries in the hash table increases, the number of collisions increases.

II. Recursive programs are efficient.

III. The worst time complexity of quick sort is $O(n^2)$.

IV. Binary search implemented using a linked list is efficient.

(a) I and II (b) II and III (c) I and IV (d) I and III

***83.** The number of binary trees with 3 nodes which when traversed in post-order gives the sequence `A,B,C` is

(a) 3 (b) 9 (c) 7 (d) 5

84. The minimum number of colors needed to color a graph having n (>3) vertices and 2 edges is

(a) 4 (b) 3 (c) 2 (d) 1

85. Which of the following file organizations is preferred for secondary key processing?

(a) Indexed sequential file organization (b) Two-way linked list

(c) Inverted file organization (d) Sequential file organization

86. Mr. Fool designed a crazy language called STUPID that included the following features.

+ has precedence over /

/ has precedence over – (binary)

– (binary) has precedence over *

* and ^ (exponentiation) have the same precedence.

+ and * associate from right to left.

The rest of the mentioned operators associate from left to right. Choose the correct stack priorities Mr. Fool should assign to `+, *, ^, /` respectively, for correctly converting an arithmetic expression in infix form to the equivalent postfix form.

(a) 5, 1, 2, 4 (b) 5, 5, 2, 4 (c) 1, 1, 2, 4 (d) 5, 4, 3, 1

87. The infix priorities of `+, *, ^, /` could be

(a) 5, 1, 2, 7 (b) 7, 5, 2, 1 (c) 1, 2, 5, 7 (d) 5, 2, 2, 4

88. Mr. Fool's STUPID language will evaluate the expression `2 * 2 ^ 3 * 4` to

(a) 256 (b) 64 (c) 4^{12} (d) 4^{81}

89. The expression `1 * 2 ^ 3 * 4 ^ 5 * 6` will be evaluated to

(a) 32^{30} (b) 162^{30} (c) 49152 (d) 173458

90. In a circularly linked list organization, insertion of a record involves the modification of

(a) no pointer (b) 1 pointer (c) 2 pointers (d) 3 pointers

91. Stack is useful for implementing

(a) radix sort (b) breadth first search

(c) recursion (d) depth first search

***92.** To store details of an employee, a storage space of 100 characters is needed. A magnetic tape with a density of 1000 characters per inch and an inter-record gap of 1 inch is used to store information about all employees in the company. What should be the blocking factor so that the wastage does not exceed one-third of the tape?

(a) 0.05 (b) 20 (c) 10 (d) 0.1

***93.** A machine needs a minimum of 100 sec to sort 1000 names by quick sort. The minimum time needed to sort 100 names will be approximately

(a) 50.2 sec (b) 6.7 sec (c) 72.7 sec (d) 11.2 sec

***94.** A machine took 200 sec to sort 200 names, using bubble sort. In 800 sec, it can approximately sort

(a) 400 names (b) 800 names (c) 750 names (d) 800 names

***95.** The correct order of arrangement of the names Bradman, Lamb, May, Boon, Border, Underwood and Boycott, so that the quicksort algorithm makes the least number of comparisons is

(a) Bradman, Border, Boon, Boycott, May, Lamb, Underwood

(b) Bradman, Border, Boycott, Boon, May, Underwood, Lamb

(c) Underwood, Border, Boon Boycott, May, Lamb, Bradman

(d) Bradman, May, Lamb, Border, Boon, Boycott, Underwood

***96.** Which of the following is useful in traversing a given graph by breadth first search?

(a) Stack (b) Set (c) List (d) Queue

97. Which of the following is useful in implementing quick sort?

(a) Stack (b) Set (c) List (d) Queue

98. Queue can be used to implement

(a) radix sort (b) quick sort (c) recursion (d) depth first search

***99.** The expression tree given in Fig. 9.10 evaluates to 1, if

(a) $a = -b$ and $e = 0$ (b) $a = -b$ and $e = 1$

(c) $a = b$ and $e = 0$ (d) $a = b$ and $e = 1$

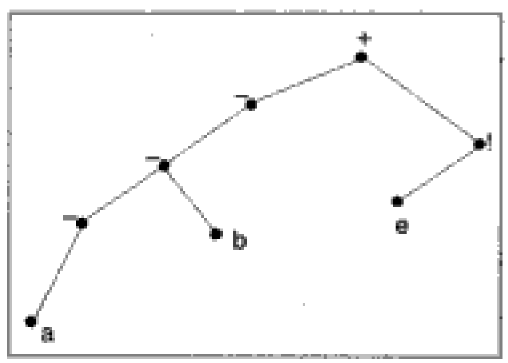


Fig. 9.10

***100.** A hash function randomly distributes records one by one in a space that can hold x number of records. The probability that the m^{th} record is the first record to result in collision is

(a) $(x-1)\ (x-2)\ldots(x-(m-2))(m-1) / x^{m-1}$

(b) $(x-1)\ (x-2)\ldots(x-(m-1))(m-1) / x^{m-1}$

(c) $(x-1)\ (x-2)\ldots(x-(m-2))(m-1) / x^{m}$

(d) $(x-1)\ (x-2)\ldots(x-(m-1))(m-1) / x^{m}$

101. The process of accessing data stored in a tape is similar to manipulating data on a

(a) stack (b) queue (c) list (d) heap

102. If the hashing function is the remainder on division, then clustering is more likely to occur if the storage space is divided into 40 sectors rather than 41. This conclusion is

(a) more likely to be false (b) more likely to be true

(c) is always false (d) none of the above

103. Unrestricted use of `goto` is harmful, because it

(a) makes debugging difficult

(b) increases the running time of programs

(c) increases memory requirement of programs

(d) results in the compiler generating longer machine code

104. The maximum degree of any vertex in a simple graph with n vertices is

(a) n (b) n–1 (c) n+1 (d) 2n–1

105. The recurrence relation that arises in relation with the complexity of binary search is

(a) $T(n) = T(n/2) + k$, where k is a constant

(b) $T(n) = 2T(n/2) + k$, where k is a constant

(c) $T(n) = T(n/2) + \log(n)$

(d) $T(n) = T(n/2) + n$

106. An item that is read as input can be either pushed to a stack and later popped and printed, or printed directly. Which of the following will be the output if the input is the sequence of items - `1,2,3,4,5`?

(a) `3, 4, 5, 1, 2` (b) `3, 4, 5, 2, 1`

(c) `1, 5, 2, 3, 4` (d) `5, 4, 3, 1, 2`

107. Which of the following algorithm design technique is used in the quick sort algorithm?

(a) Dynamic programming (b) Backtracking

(c) Divide and conquer (d) Greedy method

108. Linked lists are not suitable for implementing

(a) insertion sort (b) binary search

(c) radix sort (d) polynomial manipulation

109. Which one of the following statements is false?

(a) Optimal binary search tree construction can be performed efficiently using dynamic programming.

(b) Breadth-first search cannot be used to find connected components of a graph.

(c) Given the prefix and postfix walks of a binary tree, the binary tree cannot be uniquely reconstructed.

(d) Depth-first search can be used to find the connected components of a graph.

***110.** The number of edges in a regular graph of degree d and n vertices is

(a) maximum of n, d (b) $n+d$ (c) nd (d) $nd/2$

111. Consider the following two functions.

$$f(n) = n^3, \text{ if } 0 \le n < 10{,}000$$
$$n^2, \text{ otherwise}$$
$$g(n) = n, \text{ if } 0 \le n < 100$$
$$n^2+5n, \text{ otherwise}$$

Which of the following is/are true?

(a) $f(n)$ is $O(n^3)$ (b) $g(n)$ is $O(n^3)$

(c) $O(f(n))$ is same as $O(g(n))$ (d) $g(n)$ is $O(n^2)$

***112.** A 3-ary tree is a tree in which every internal node has exactly 3 children. The number of leaf nodes in such a tree with 6 internal nodes will be

(a) 10 (b) 23 (c) 17 (d) 13

113. The concatenation of two lists is to be performed in O(1) time. Which of the following implementations of a list could be used?

(a) Singly linked list (b) Doubly linked list

(c) Circular doubly linked list (d) Array implementation of list

114. The correct matching for the following pairs is

(A) All pairs shortest path (1) Greedy

(B) Quick sort (2) Depth-first search

(C) Minimum weight spanning tree (3) Dynamic programming

(D) Connected Components (4)Divide and conquer

(a) A-2, B-4, C-1, D-3 (b) A-3, B-4, C-1, D-2

(c) A-3, B-4, C-2, D-1 (d) A-4, B-1, C-2, D-3

115. Which of the following is essential for converting an infix expression to the postfix form efficiently?

(a) An operator stack (b) An operand stack

(c) An operator stack and an operand stack (d) A parse tree

***116.** A binary search tree contains the values – 1, 2, 3, 4, 5, 6, 7, and 8. The tree is traversed in preorder and the values are printed out. Which of the following sequences is a valid output?

(a) 5 3 1 2 4 7 8 6 (b) 5 3 1 2 6 4 9 7

(c) 5 3 2 4 1 6 7 8 (d) 5 3 1 2 4 7 6 8

117. Let T(n) be the function defined by

$$T(1) = 1, \text{ if } n = 1$$
$$= 2T(\lfloor n/2 \rfloor) + \sqrt{n}, \text{ for } n \ge 2$$

Which of the following statements is true?

(a) $T(n) = O(\sqrt{n})$ (b) $T(n) = O(n)$

(c) $T(n) = O(\log n)$ (d) None of the above

118. Which of the following need not be a binary tree?

(a) Search tree (b) Heap (c) AVL-Tree (d) B-Tree

***119.** Assume 5 buffer pages are available to sort a file of 105 pages. The cost of sorting using m-way merge sort is

(a) 206 (b) 618 (c) 840 (d) 926

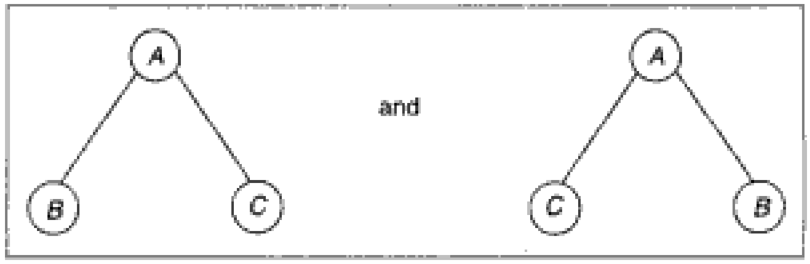


Fig. 9.11

This gives us 6 more possibilities.

6. Each comparison puts one element in the final sorted array. In the worst case $m+n-1$ comparisons are necessary.

8. It will be one more than the size of the biggest cluster (which is 4) in this case. This is because, assume a search key hashing onto bin 8. By linear probing the next location for searching is bin 9. Then 0, then 1. If all these resulted in a miss, we try at bin 2 and stop as it is vacant. This logic may not work if deletion operation is done before the search.

9. A strictly binary tree with 'n' leaves must have $(2n-1)$ nodes. Verify for some small 'n'. This can be proved by the principle of mathematical induction.

10. If the depth is d, the number of nodes n will be $2^{(d+1)}-1$.
So, $n+1 = 2^{(d+1)}$ or $d = \log(n+1) - 1$

12. For example, consider the binary search tree given in Qn.2. The inorder listing will be 1, 4, 5, 6, 10, 11, 12, 30, i.e. the numbers arranged in ascending order.

13. The 10 items $i1$, $i2$, ... $i10$ may be arranged in a binary search tree as in Fig. 9.12. To much $i5$, the number of comparison needed is 1; for $i2$, it is 2; for $i8$ it is 2; for $i1$ it is 3, and so on. The average is $(1+(2+2)+(3+3+3+3)+(4+4+4))/10$, i.e., 2.9.

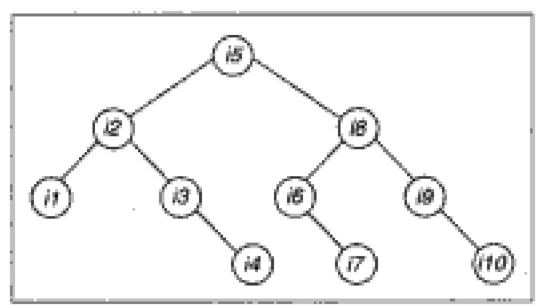


Fig. 9.12

14. `f(37) = 37 mod 7 = 2`. So, `37` will be put in location 2. `f(38) = 3`. So, `38` will be in third location. `f(72) = 2`. This results in a collision. With linear probing as the collision resolving strategy, the alternate location for `72` will be the location 4 (i.e. next vacant slot in the current configuration). Continuing this way, the final configuration will be `98, 56, 37, 38, 72, 11, 48`.

15. If the search key matches the very first item, with one comparison we can terminate. If it is second, two comparisons, etc. So, average is $(1+2+\ldots+n)/n$, i.e., $(n+1)/2$

17. Let m > n. Let m/n yield a quotient x and remainder y. So, m = n*x+y and y < m div 3 is the quotient when m is divided by 3. So, that many times p is added, before we terminate recursion by satisfying the end condition Q(m, n) = 0, if m < n. Hence the result.

18. Since the access is sequential, greater the distance, greater will be the access time. Since all the files are referenced with equal frequency, overall access time can be reduced by arranging them as in option (a).

19. Refer Qn. 18. Since each file is referenced with equal frequency and each record in a particular file can be referenced with equal frequency, average access time will be (25+(50+40)+(50+80+50)+...)/6 = 268 (approximately).

20. By definition of order, there exists constants c1, c2, n1, n2 such that

$T(n) \leq c1 \times f(n)$, for all $n \geq n1$.

$T(n) \leq c2 \times g(n)$, for all $n \geq n2$.

Let $N = \max(n1, n2)$ and $C = \max(c1, c2)$. So,

$T(n) \leq C \times f(n)$ for all $n \geq N$

$T(n) \leq C \times g(n)$ for all $n \geq N$

Adding $T(n) \leq C/2 \times (f(n)+g(n))$

Without loss of generality, let $\max(f(n), g(n)) = f(n)$.

So, $T(n) \leq C/2\ (f(n)+ f(n)) \leq C \times f(n)$.

So, order is f(n), which is max(f(n), g(n)), by our assumption.

22. By recursively applying the relation we finally arrive at

$$T(n-1) = c(n-1) + T(1) = c(n-1)+d$$

So, order is *n*.

24. The five possible trees are

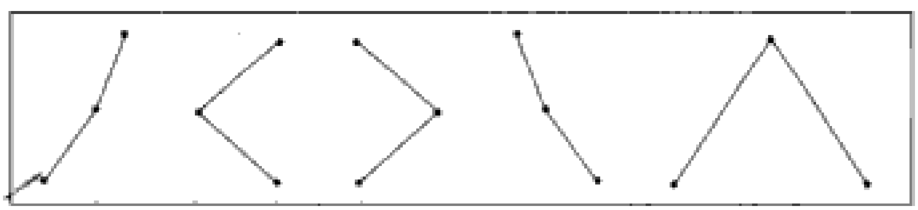


Fig. 9.13

25. Eight possible trees of depth 3. Six possible trees of depth 2. Altogether 14.

30. Using Hoffman's algorithm, code for a is 1111; b is 0; c is 110; d is 1110; e is 10. Average code length

is $4 \times .12 + 1 \times .4 + 3 \times .15 + 4 \times .08 + 2 \times .25 = 2.15$

32. Let us find what is $T(4)$, $T(5)$, $T(6)$.

$$T(4) = T(3) + T(2) - T(1) = 3 + 2 - 1 = 4$$

$$T(5) = T(4) + T(3) - T(2) = 4 + 3 - 2 = 5$$

$$T(6) = T(5) + T(4) - T(3) = 5 + 4 - 3 = 6$$

By induction it can be proved that $T(n) = n$. Hence order is *n*.

33. Refer Qn.32. Let $T(1) = T(2) = T(3) = k$ (say). Then $T(4) = k + k - k = k$

$$T(5) = k + k - k = k.$$

By mathematical induction it can be proved that $T(n) = k$, a constant.

35. In the worst case it has to check all the 2^n possible input combinations, which is exponential.

36. The postfix equivalent is 2 3 * 4 5 + −. For evaluating this using stack, starting from the left, we have to scan one by one. If it is an operand push. If it is an operator, pop it twice, apply the operator on the popped out entries and push the result onto the stack. If we follow this, we can find configuration in option (d) is not possible.

39. Merge-sort combines two given sorted lists into one sorted list. For this problem let the final sorted order be -*a*, *b*, *c*, *d*. The two lists (of length two each) should fall into one of the following 3 categories.

(i) *a*, *b* and *c*, *d* (ii) *a*, *c* and *b*, *d* (iii) *a*, *d* and *b*, *c*

The number of comparisons needed in each case will be 2,3,3. So, average number of comparisons will be (2 + 3 + 3)/3 = 8/3

Here is a better way of doing:

Let list *L*1 have the items *a*,*c* and *L*2 have the items *b*,*d*.

The tree drawn below, depicts the different possible cases. (*a*&*b* means *a* is compared with *b*. If *a* is smaller, the edge will be labeled *a*. The number within a circle, beside the leaf nodes, is the number of comparisons, needed to reach it.)

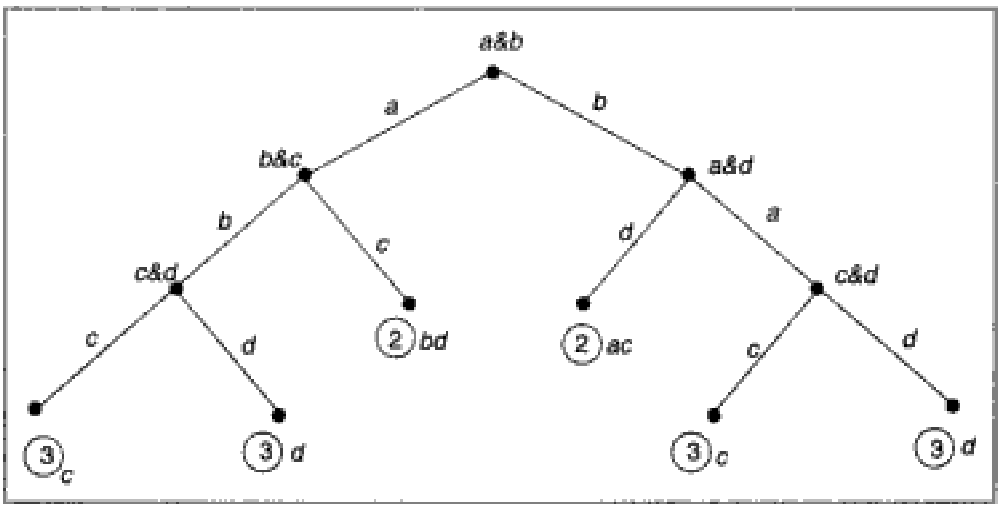


Fig. 9.14

From the tree, we find there are 6 possible ways. Total number of comparisons needed is 3+3+2+2+3+3 = 16. So, average number of comparisons is 16/6 = 8/3.

43. The maximum number of comparison is number of items × radix × number of digits i.e., 7 × 10 × 4 = 280,

47. Dijkstra's algorithm solves single source shortest path problem.

Warshall's algorithm finds transitive closure of a given graph.

Prim's algorithm constructs a minimum cost spanning tree for a given weighted graph.

48. Third row corresponds to node 3. From 3 to 1, there is no path, So, the entry (3,1) should be zero. Since there is a path from 3 to 2 and also from 3 to 3 (i.e., 3 → 2 → 3), the third row should be 0,1,1.

49. Eccentricity of a given node is the maximum of minimum path from other nodes to the given node.

Cost of minimum path from 1 to 5 is 7

Cost of minimum path from 2 to 5 is 6

Cost of minimum path from 3 to 5 is 4

Cost of minimum path from 4 to 5 is 7

Since the maximum is 7, eccentricity of node 5 is 7.

50. Refer Qn.49.

Find eccentricity of all nodes.

Eccentricity of node 1 is ∞

Eccentricity of node 2 is 6

Eccentricity of node 3 is 8

Eccentricity of node 4 is 5

Eccentricity of node 5 is 7

Center of a graph is the node with minimum eccentricity.

52. Total number of possible permutations for the previous problem is 5. For the four entries `a,b,c,d` the possibilities are `a`, followed by permutations of `a,b,c` which is 5. `b`, followed by permutations of `a,c, d,` which is 5.The other possibilities are `c,b,a,d ; c,d,b,a ; c,b,d,a ; d,c,b,a`. Totally 14.

56. Conventional way needs a storage of $m \times n$.

In the case of linked list implementation of sparse matrices, storage needed will be $m + 3 \times$ (the number of non-zero entries).

Only in case (c), both the methods need the same storage of 30.

59. The tree whose preorder traversal yields `* + A B - C D`, is given in Fig. 9.15. Write the post-order traversal of the tree. That is the post-fix form.

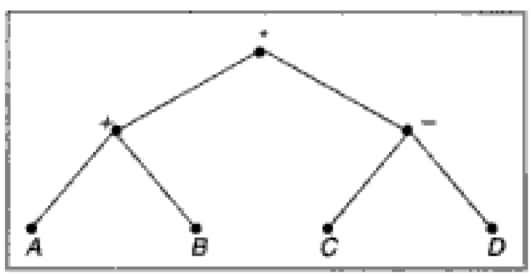


Fig. 9.15

60. Let there be 'n' items to be searched, After the first search the list is divided into two, each of length $n/2$. After the next search, 2 lists, each of length $n/4$ and so on. This successive division has to stop when the length of list becomes 1. Let it happen after k steps. After the k steps, $n/2^k = 1$. Solving, $n = 2^k$. Hence the order is $\log(n)$.

61. Load factor is the ratio of number of records that are currently present and the total number of records that can be present. If the load factor is less, free space will be more. This means probability of collision is less. So, the search time will be less.

62. If the new record hashes onto one of the six locations 7, 8, 9, 10, 1 or 2, the location 2 will receive the new record. The probability is 6/10 (as 10 is the total possible number of locations).

63. You can verify that the 1st, 3rd, 5th, 7th…probes check at location 5.

The 2nd, 6th, 10th…probes check at location 8.

The 4th, 8th, 12th…probes check at location 4.

The rest of the address space will never be probed.

64. If there is only one record, then the probability of a collision will be 1/100. If 2, then 2/100 etc., If 9 then 9/100. So, the required probability is 1 + 2 + 3 ... 9/100 = 0.45.

69. If the (1, 3) entry in M^3 is 2, it means there are 2 paths of length 3, connecting nodes 1 and 3. If you see the graph in option (a), there are 2 paths connecting 1 and 3, ($1 \to 2 \to 3 \to 3$ and $1 \to 3 \to 3 \to 3$).

70. In breadth first traversal the nodes are searched level by level. Starting with vertex `A`, the only next choice is `B`. Then `C`, then `1` and lastly `2`. Comparing with `ABCDE`, option (a) is the correct answer.

71. In the depth first traversal, we go as deep as possible before we backtrack and look for alternate branches. Here it yields `ABC21`. So, labels of nodes `1` and `2` should be `E` and `D` respectively.

73. In topological sorting we have to list out all the nodes in such a way that whenever there is an edge connecting `i` and `j`, `i` should precede `j` in the listing. For some graphs, this is not at all possible, for some this can be done in more than one way. Option (d) is the only correct answer for this question.

74. Strong component of a given graph is the maximal set of vertices such that for any two vertices *i*, *j* in the set, there is a path connecting *i* to *j*. Obviously vertex '*d*' can't be in the maximal set (as no vertex can be reached starting from vertex *d*). The correct answer is option (d).

75. Use Prim's algorithm or Kruskal's algorithm and verify the result.

78. Each comparison will append one item to the existing merge list. In the worst case one needs $m + n - 1$ comparisons which is of order $m+n$.

79. It can be proved by induction that a strictly binary tree with 'n' leaf nodes will have a total of $2n - 1$ nodes. So, number of non-leaf nodes is $(2n-1)-n = n-1$.

82. Recursive programs take more time than the equivalent non-recursive version and so not efficient. This is because of the function call overhead.

In binary search, since every time the current list is probed at the middle, random access is preferred. Since linked list does not support random access, binary search implemented this way is inefficient.

83. The 5 binary trees are

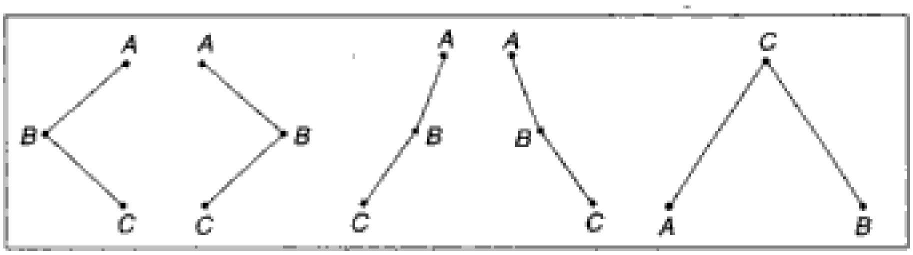


Fig. 9.16

92. Blocking factor is the number of logical records that is packed to one physical record. Here in every 3 inch, there should be 2 inch of information. Hence 2 × 10 = 20 logical records.

93. In the best case quick sort algorithm makes $n\log(n)$ comparisons. So $1000 \times \log(1000)$ = 9000 comparisons, which takes 100 sec. To sort 100 names a minimum of 100 (log 100) = 600 comparisons are needed. This takes $100 \times 600/9000 = 6.7$ sec.

94. For sorting 200 names bubble sort makes $200 \times 199/2 = 19900$ comparisons. The time needed for 1 comparison is 200 sec (approximately). In 800 sec it can make 80,000 comparisons. We have to find n, such that $n(n-1)/2 = 80,000$. Solving, n is approximately 400.

95. Let the first element be the pivot element always. The best way is the one that splits the list into two equal parts each time. This is possible if the pivot element is the median. Consider the given set of names or the equivalent set 1, 2, 3, 4, 5, 6, 7. Four is the median and hence should be the pivot element. Since the first element is the pivot element, 4 should be the first element. After the first pass, 4 will be put in the correct place and we are left with two sub lists 1, 2, 3 and 5, 6, 7. Since 2 is the median of 1, 2, 3 the list should be rearranged as 2, 1, 3 or 2, 3, 1. For similar reasons 5, 6, 7 should be rearranged as 6, 5, 7 or 6, 7, 5.

96. Immediately after visiting a node, append it to the queue. After visiting all its children, the node currently in the head of the queue is deleted. This process is recursively carried out on the current head of the queue, till the queue becomes empty.

99. The corresponding expression is `-(-a-b)+e!`. This is `1` if `a=-b` and `e` is either `1` or `0`, since `1!=0!=1`.

100. Probability for the first record not colliding is x/x.

Probability for the second record not colliding is $x - 1/x$.

(This is because one place is already occupied. So, favorable number of cases is $x-1$). Probability for the third record not colliding is $x - 2/x$.

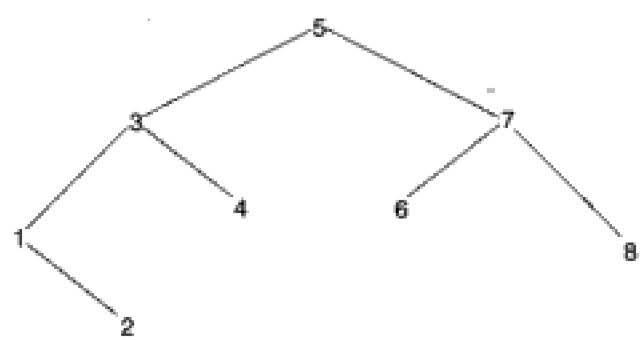
Probability for the $(m-1)^{th}$ record not colliding is $x - (m-2)/x$.

Now the next (m^{th}) record is resulting in a collision. Out of the x places, it should hash to one of the $(m-1)$ places already, filled. So probability is $(m-1)/x$. The required probability is $(x/x)\ (x - 1/x)\ (x - 2/x) \ldots (x - (m - 2)/x)\ (m - 1/x)$

110. In a regular graph, all the vertices will be of the same degree. Total degrees of all the vertices is nd. Each edge will be increasing the total degree by 2. So, totally $nd/2$ edges.

112. It can be proved by induction that any 3-ary tree with n internal nodes will have exactly $2(n-1) + 3$ leaf nodes. In this question n is 6.

116. The tree for option (d) is:



For the other options it is impossible to construct a binary search tree having the listed preorder.