CSD 102 Data Structures

End-Semester Exam

Date: 03-12-2024 Total Marks = 60 Time: 09:00 AM - 11:00 AM

Exam Instructions

1. You are **not permitted** to carry any gadget for communication/computation (including **mobile phones, scientific calculator, and smart watches**) inside the examination room.
2. It is **mandatory** for you to carry the university ID card for being permitted to take the examination. In the event of loss of the ID card, you are advised to contact Mr Gaurav Paliwal and obtain a duplicate ID card to enable you to write the examination. The time lost, if any, in the transaction for obtaining a duplicate card **shall not** be compensated for, during the examination.
3. You shall not be allowed entry to the examination room **10** minutes after the scheduled commencement of the examination, and shall not be permitted to leave the room prior to **30** minutes after the commencement of the examination.
4. It is **mandatory** to write your **roll number** on the **question paper** and **answer sheet**, as soon as you receive them.
5. Only **one** student at a time will be permitted to leave the examination hall for using the toilet.

You are advised to strictly comply with the above instructions.

Name: _____

Roll No: _____

Room No: _____

Student Signature: _____

Invigilator Signature: _____

Objective

1. Given the following sequence: {2, 3, 5, 6, 9, 11, 15}. Which sorting algorithm will run in O(n) time (n comparisons)?
    a. Insertion Sort
    b. Selection Sort
    c. Heap Sort
    d. Merge Sort **[1 Mark]**

Answer: a

2. A doubly linked list is declared as:

```
struct Node {
  int Value;
  struct Node *Fwd;
  struct Node *Bwd;
};
```

[1 Mark]

Where Fwd and Bwd represent forward and backward link to the adjacent elements of the list. Which of the following segment of code deletes the node pointed to by X from the doubly linked list, if it is assumed that X points to neither the first nor the last node of the list?

a. X→ Bwd.Fwd =X→ Fwd; X. Fwd → Bwd =X→ Bwd;
b. X. Bwd → Fwd =X. Bwd; x→Fwd.Bwd=X.Bwd
c. X→ Bwd → Fwd =*X→* Fwd; *X→Fwd→Bwd=X→Bwd*;
d. X→ Bwd → Fwd =X→ Bwd; X→ Fwd → Bwd =X→ Fwd;

Answer: c

3. Which of the following statements is true?
    a. As the number of entries in a hash table increases, the number of collisions increases.
    b. Recursive programs are efficient
    c. The worst-case complexity for Quicksort is $O(n^2)$
    d. Binary search using a linear linked list is efficient **[1 Mark]**
        a. I and II
        b. II and III
        c. I and IV
        d. I and III

Answer: d

4. Program P reads in 500 integers in the range [0..100] representing the scores of 500 students. It then prints the frequency of each score above 50. What would be the best way for P to store the frequencies?
    a. An array of 50 numbers
    b. An array of 100 numbers
    c. An array of 500 numbers

d. A dynamically allocated array of 550 numbers **[1 Mark]**

Answer: a

The below function will store the frequency of score > 50

```
void calculateFrequency(int nums[ ], int n)  {
    // n is the size of student marks arrray
    // nums is the array containing marks
    int freq [50]; // initializ the array with 0
    for(int i = 0; i <  n; i++) {
        freq[nums[i] - 51]++;
    }
}
```

Index 0 will contains a frequency of 51 Marks.

Index 1 will contains a frequency of 52 Marks.

:

:

Index 49 will contains a frequency of 100 Marks.

Therefore an array of 50 numbers is sufficient

5. Find the complexity of the below program:

```
void function(int n)
{
  int count = 0;
  for (int i=n/2; i<=n; i++)
    for (int j=1; j+n/2<=n; j = j++)
      for (int k=1; k<=n; k = k * 2)
        count++;
}
```

[2 Marks]

a. $O(n\log^2 n)$
b. $O(n)$
c. $O(n^2 \log n)$
d. $O(n^2)$

Answer: c. Time Complexity: $O(n^2 \log n)$.

```c
void function(int n)
{
    int count = 0;

    // outer loop executes n/2 times
    for (int i=n/2; i<=n; i++)

        // middle loop executes  n/2 times
        for (int j=1; j+n/2<=n; j = j++)

            // inner loop executes logn times
            for (int k=1; k<=n; k = k * 2)
                count++;
}
```

6. Following is C like pseudo-code of a function that takes a Queue as an argument and uses a stack S to do processing.

```c
void fun (Queue *Q)
{
    Stack S;  // Say it creates an empty stack S
    // Run while Q is not empty
    while (!isEmpty(Q))
    {
        // deQueue an item from Q and push the dequeued item to S
        push(&S, deQueue(Q));
    }
    // Run while Stack S is not empty
    while (!isEmpty(&S))
    {
      // Pop an item from S and enqueue the popped item to Q
      enQueue(Q, pop(&S));
    }
}
```

What does the above function do in general?

a. Removes the last from Q
b. Keeps the Q same as it was before the call
c. Makes Q empty
d. Reverses the Q **[2 Marks]**

Answer: d

7. The preorder traversal of a binary search tree is 15, 10, 12, 11, 20, 18, 16, 19. Which one of the following is the postorder traversal of the tree?
    a. 10, 11, 12, 15, 16, 18, 19, 20
    b. 11, 12, 10, 16, 19, 18, 20, 15
    c. 20, 19, 18, 16, 15, 12, 11, 10
    d. 19, 16, 18, 20, 11, 12, 10, 15 **[2 Marks]**

Answer: b

8. Which of the following runs fastest?
    a. Searching a node in an AVL tree
    b. Searching a node in a Binary Search Tree (all items are identical)
    c. Running Pre-order traversal algorithm on a BST
    d. Searching for an element in a circular linked list **[2 Marks]**

Answer: a

Subjective

9. Consider the following C program.

```
#include < stdio.h >
int main () {
  int a [4] [5] = {{1, 2, 3, 4, 5},
                   {6, 7, 8, 9, 10},
                   {11, 12, 13, 14, 15},
                   {16, 17, 18, 19, 20}};
  printf ("%d\n", *(*(a+**a+2) +3));
  return (0);
}
```

The output of the program is _____. **[2 Marks]**

Answer: 19

10. Consider the operator precedence and associativity rules for the integer arithmetic operators given in the table below. **[2 Marks]**

Operator	Precedence	Associativity
+	Highest	Left
-	High	Right
*	Medium	Right
/	Low	Right

The value of the expression 3+1+5∗2/7+2−4−7−6/2 as per the above rules is _____.

Answer: 6

The given expression is evaluated as follows:

$\Longrightarrow \underbrace{3+1}+5*2/\underbrace{7+2}-4-7-6/2$ [As '+' has higher predence with left associative in the operators present in the expression.]

$\Longrightarrow (3+1)+5*2/7+2-4-7-6/2$

$\Longrightarrow (4+5)*2/7+2-4-7-6/2$

$\Longrightarrow 9*2/(7+2)-4-7-6/2$

$\Longrightarrow 9*2/9-4-\underbrace{7-6}/2$ [As '-' has higher predence with right associative in the operators present in the expression.]

$\Longrightarrow 9*2/9-4-(7-6)/2$

$\Longrightarrow 9*2/9-(4-1)/2$

$\Longrightarrow 9*2/(9-3)/2$

$\Longrightarrow \underbrace{9*2}/6/2$ [As '*' has higher predence with right associative in the operators present in the expression.]

$\Longrightarrow (9*2)/6/2$

$\Longrightarrow 18/\underbrace{6/2}$ [As '/' has higher predence with right associative in the operators present in the expression.]

$\Longrightarrow 18/(6/2)$

$\Longrightarrow 18/3=6$

11. List the following functions by increasing asymptotic growth rate. **[2 Marks]**

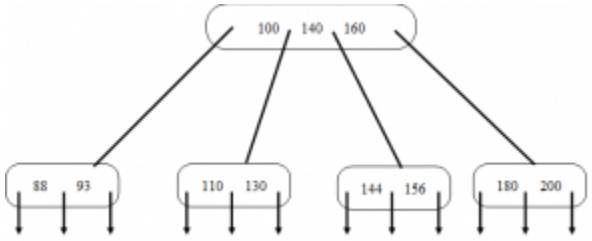
$$\lg n \qquad 1.1^n \qquad n\lg n \qquad n(\lg n)^2 \qquad 3\lg n \qquad 2^5 \qquad n^{34}$$

Answer:

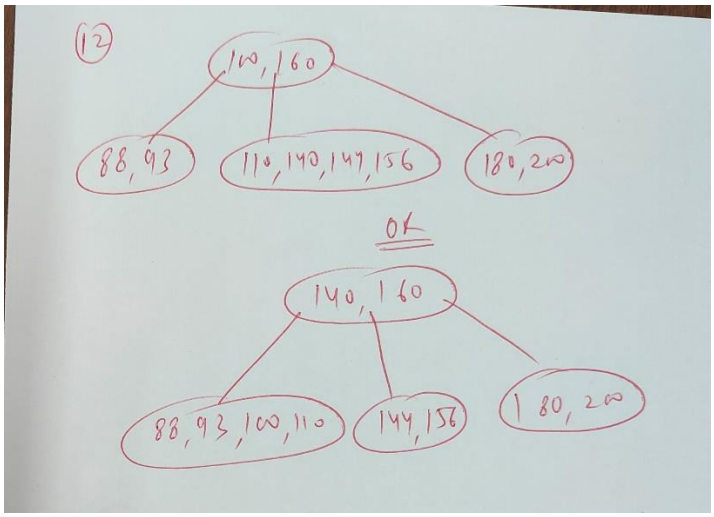
$$2^5 \qquad \underbrace{\lg n \qquad 3\lg n}_{\text{Same asymptotic behaviour}} \qquad n\lg n \qquad n(\lg n)^2 \qquad n^{34} \qquad 1.1^n$$

By discarding constant coefficients we see that $3\lg n = \Theta(3\lg n) = \Theta(\lg n)$.

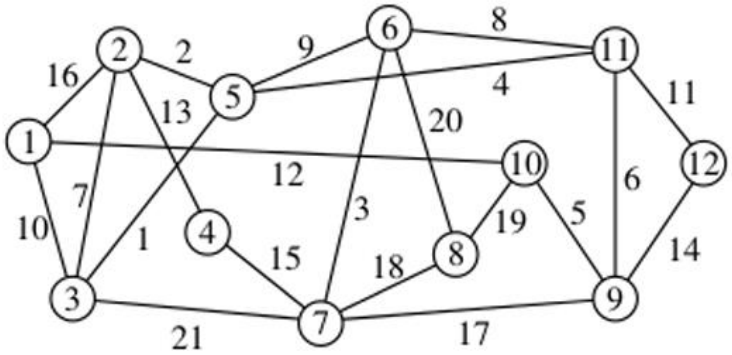
12. The figure shown below is B-tree of order 5. Draw the final B-tree after deleting 130 from the tree? **[2 Marks]**



Answer:



13. Consider the below graph. If we start with node 10 as the starting node and use Prim's algorithm to construct the minimum spanning tree.
    a. Give the order in which nodes are covered while constructing MST. **[3 Marks]**
    b. Give the minimum total weight. **[3 Marks]**



Answer: The order is 10, 9, 11, 5, 3, 2, 6, 7, 1, 12, 4, 8. Total weight 81.

14. What would be the breadth-first search (BFS) and depth-first search (DFS) of the above graph while considering the starting vertex as 5? If there is ever a decision between multiple neighbor nodes in the BFS and DFS algorithms, assume we always choose the integer closest to 0. **[3 Marks (BFS) + 3 Marks (DFS) = 6 Marks]**

Answer: BFS: 5, 2, 3, 6, 11, 1, 4, 7, 8, 9, 12, 10

DFS: 5, 2, 1, 3, 7, 4, 6, 8, 10, 9, 11, 12

15. Consider a hash table with 9 slots. The hash function is h(k) = k mod 9. The collisions are resolved by *chaining*. Draw the final Hash table while inserting the following 9 keys in the order: 5, 28, 19, 15, 20, 33, 12, 17, 10. What is the maximum, minimum, and average chain lengths in the hash table, respectively? **[5 Marks]**

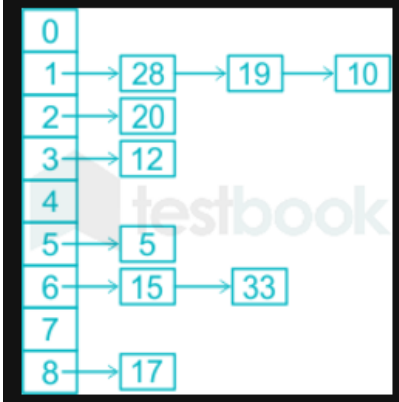
Answer:

3, 0, and 1

Keys: 5, 28, 19, 15, 20, 33, 12, 17,

10.

$h(k) = k \bmod 9$



Chaining

Maximum chain length = 3 (28 -> 19 -> 10)

Minimum chain length = 0 (0, 4, 7 slot doesn't have any element)

Average chain length $= \frac{0+3+1+1+0+1+2+0+1}{9} = \frac{9}{9} = 1$

16.

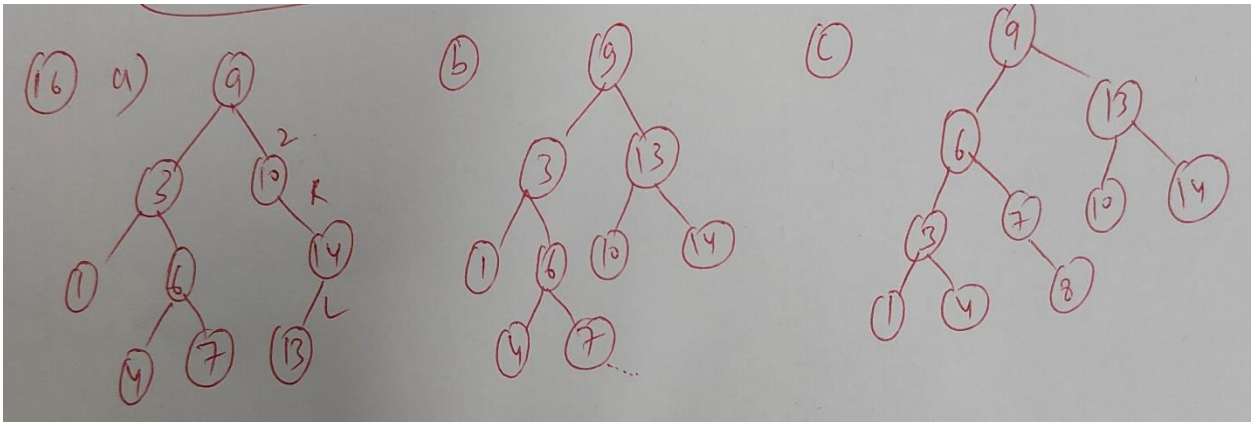
a. Draw the binary tree given in this array. **[2 Marks]**

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13
element	**9**	**3**	**10**	**1**	**6**		**14**			**4**	**7**			**13**

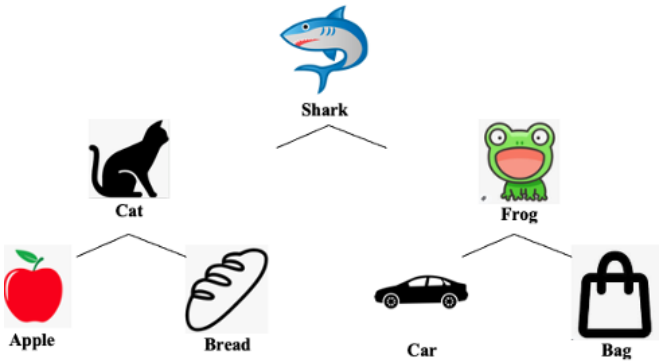
b. What do you have to do to convert this tree to AVL? Show the tree after appropriate rotations. Identify what type of rotation? **[2 Marks]**

c. Now insert a new node (8) in this tree. Show the tree after appropriate rotations. Identify what type of rotation? **[2 Marks]**

Answer:



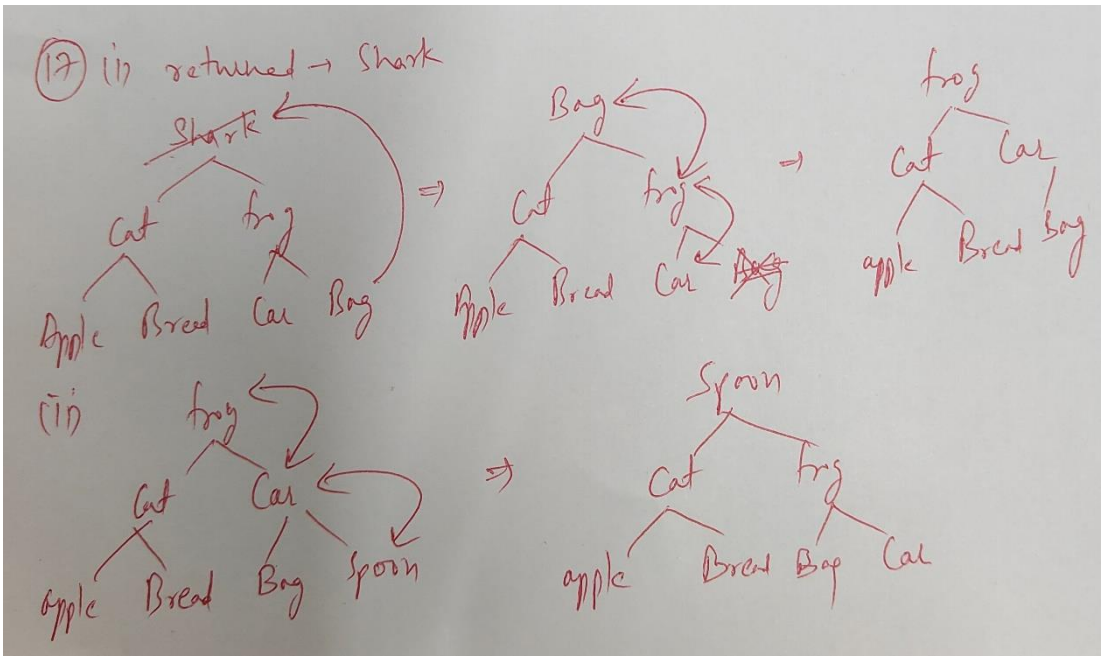
17. We need to store images in an efficient data structure. The following is a *max-heap* of images. Each image is annotated with a String text-label which is to be used to compare items. (example: "BAG" comes before "BAT").



i) Remove an image from this heap. What image is returned? Show the resulting heap. **[1 Marks]**

ii) Add  (Spoon) to this heap. Show the resulting heap. **[1 Marks]**

Answer:



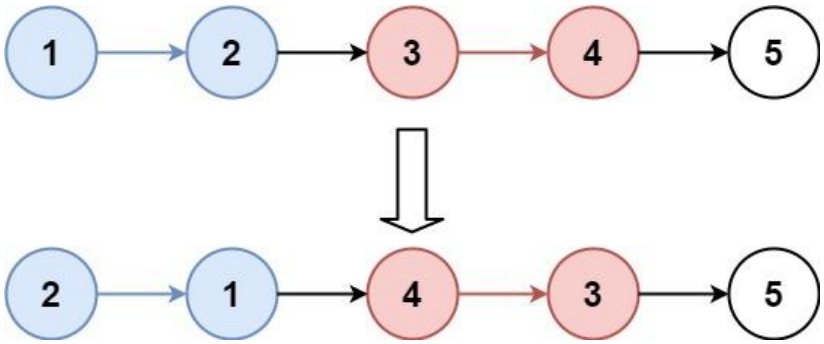
18. Reverse Nodes in k-Group

Given the *head* of a linked list, reverse the nodes of the list *k* at a time, and return the modified list.

k is a positive integer and is less than or equal to the length of the linked list. If the number of nodes is not a multiple of *k* then left-out nodes, in the end, should remain as it is. **[5 Marks]**

Note: You may not alter the values in the list's nodes, only nodes themselves may be changed. Can you solve the problem in O(1) extra memory space?

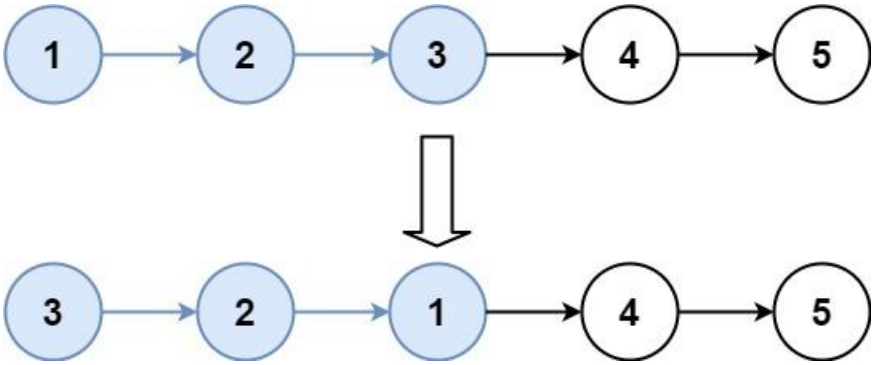
Example 1:



Input: head = [1,2,3,4,5], k = 2

Output: [2,1,4,3,5]

Example 2:



Input: head = [1,2,3,4,5], k = 3
Output: [3,2,1,4,5]

Answer: refer this link for multiple solutions: https://www.geeksforgeeks.org/reverse-a-linked-list-in-groups-of-given-size/