Write a sort note on the following

i) The following sequence of operation is performed on a stack:

push(1) , push(2) ,pop , push(1) ,push(2), pop ,pop,pop,push(2),pop

determine the sequences of popped out values.

Solution- let assume the sequence **(**2,2,1,1,2) fill into stack.

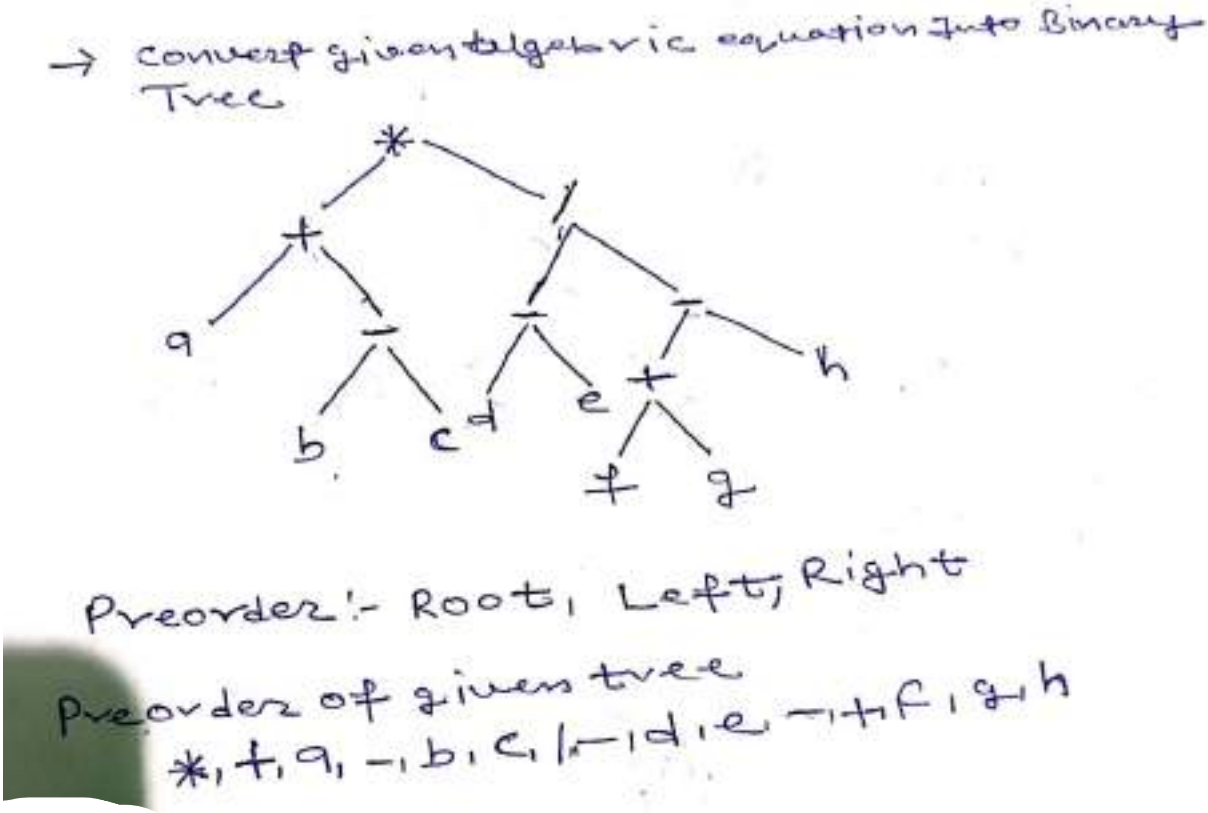
Pop sequence can be like this

Operation	**stack**	**pop sequence**
Push 1	**1**	
Push 2	**1,2**	
Pop	**1**	**2**
Push 1	**1,1**	
Push 2	**1,1,2**	
Pop	**1,1**	**2,2**
Pop	**1**	**2,2,1**
Pop	**Empty**	**2,2,1,1**
Push 2	**2**	
Pop	**Empty**	**2,2,1,1,2**

ii) Write the preorder expression of the following algebraic expression

[a+(b-c)]*[(d-e)/(f+g-h)]

Answer-



iii)Arrange the following complexity function in increasing order.

$O(n^2)$, $O(n^3)$, O(nlog), $O(2^n)$, O(log n) , O(1), $O(n^4)$, $O(3^n)$, $O(\sqrt{n})$, $O(n^n)$

Answer- $O(1) < O(\log n) < O(\sqrt{n}) < O(n\log) < O(n^2) < O(n^3) < O(n^4) < O(2^n) < O(3^n) < O(n^n)$

iv)What is the run time compexity of quick sort when the list is already sorted.

Answer-O(nlogn) [it also depend on pivot selection]

v)What is garbage collection .

answer- Garbage collection is the systematic recovery of pooled computer storage that is being used by a program when that program no longer needs the storage. ... Garbage collection is an automatic memory management feature in many modern programming languages, such as Java and languages in the . NET framework.

How we can reduce the no of bits while representing the data which will result in saving of space while storing data and saving transmission time.conclude your answer by solving the mention string.

String: a a b a a c d a a b b b d e a a a a a b b d d c c c d c a a b b c a a a b b a a b b a a

How many bits we required by using 8 bit ASCII Code , by using own code , and by using huffman,s algorithm.

Answer.

Huffman coding

→ This is used for data compression. It means using Huffman's code we can reduce no of bits representing the data, which will result in saving of space while storing data and saving transmission time and bandwidth in data transmission.

→ Huffman's coding we get a variable length code.

Example

a a b a a c d a a b b b d e a a a a a b b d d c c c d c a a b b c a a a b b a a b b a a.

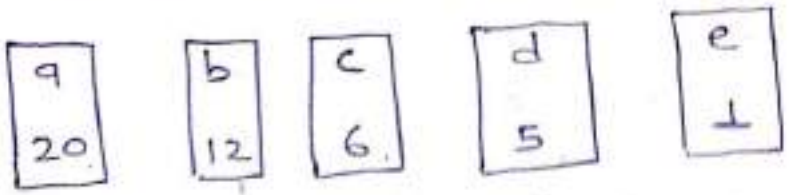
Char	frequency	No. of bits using 8 bit ASCII code
a	20.	20*8 = 160
b	12.	12*8 = 96
c	6.	6*8 = 48
d	5.	5*8 = 40
e	1	1*8 = 8
	Total	352 bits

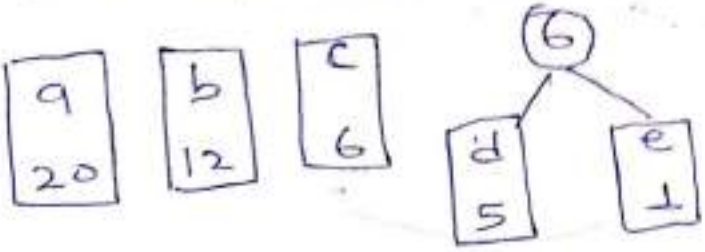
★ As in above text has only 5 character so we can make our own 5 bit code as shown below

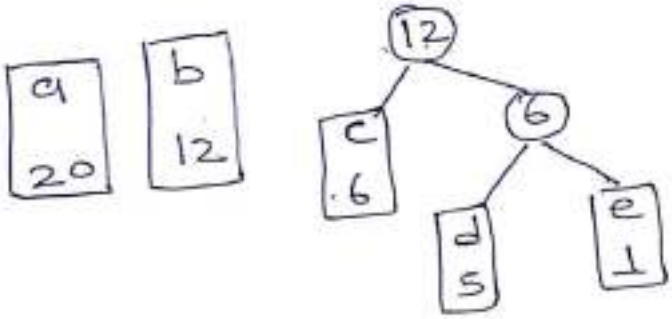
a - 0 0 0
b - 0 1 0
c - 0 1 1
d - 1 0 0
e - 1 0 1.

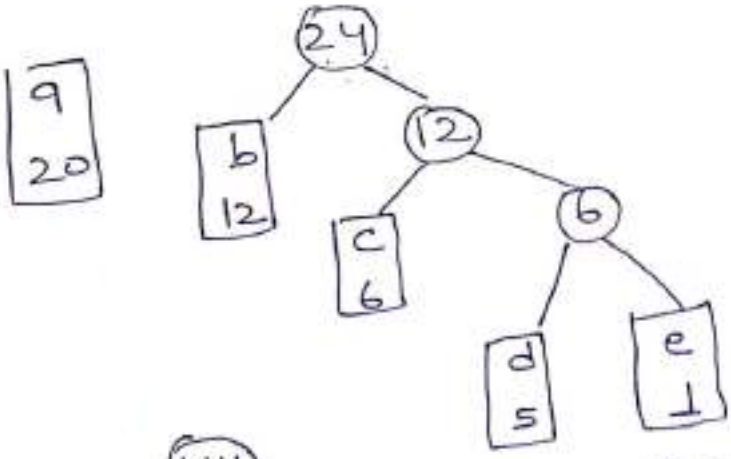
if we do so then total No. of bits require is equal to 44*3 = 132 bits

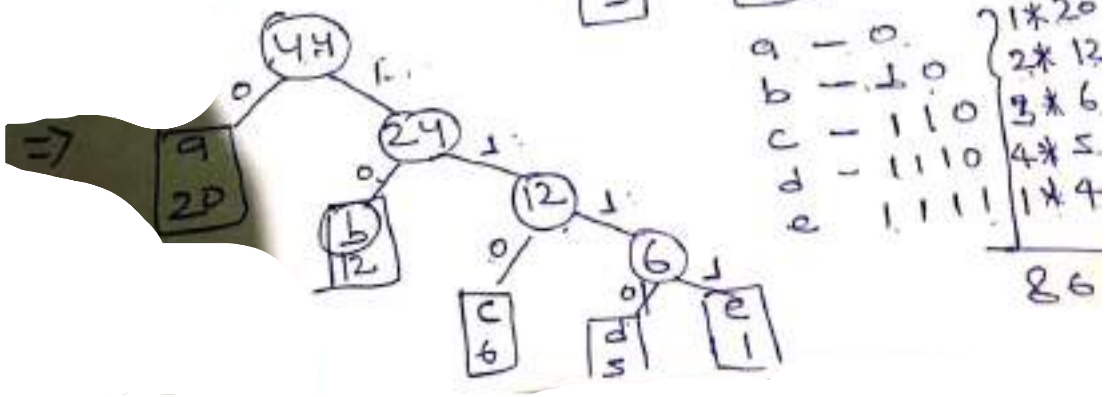
★ Now Let us make a Huffman ~~code~~ tree by considering frequency of characters in given text











a – 0
b – 10
c – 110
d – 1110
e 1111

1*20
2*12
3*6
4*5
1*4

86

$1*20 + 2*12 + 3*6 + 4*5 + 4*1 = 86$

No. of bit required

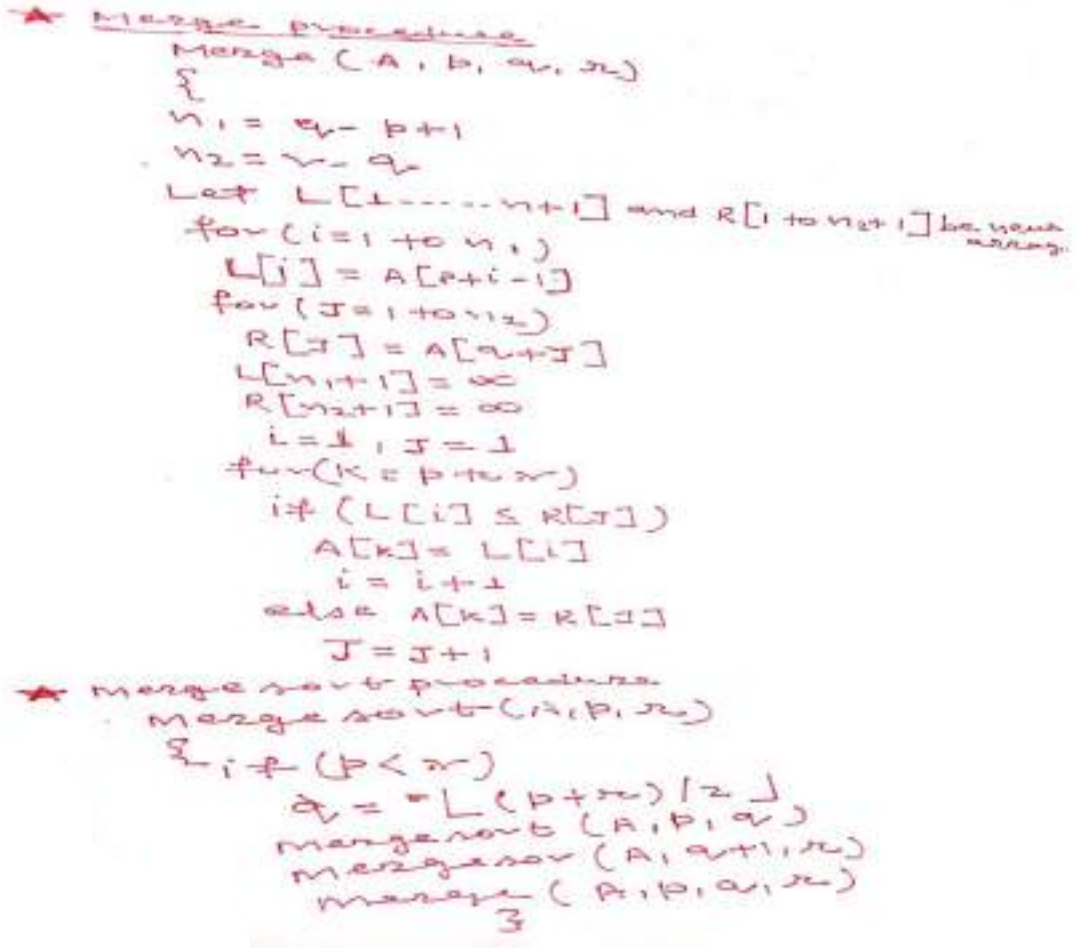
using 8 bit ASCII code = 352

using our own code = 132

using Huffman's also = 86

. **Suppose we have an array of numbers a[1],……,a[n] in which the first i number a[1],……,a[i] has been sorted into ascending order , and remaining numbers a[i+1],……..,a[n] has been in descending order.The aim is to sort the entire array in ascending order, design an algorithm which merges the two sorted lists into a second array b[1],……...,b[n] and copies the result back into a. show that the algorithm takes time O(n).**

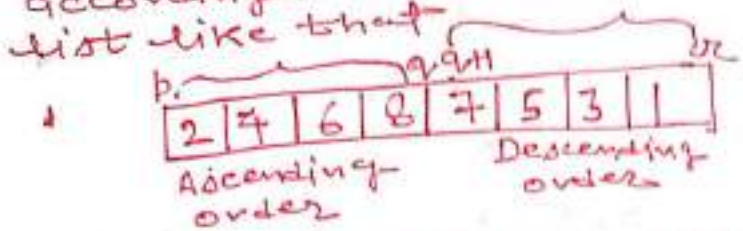
Answer.



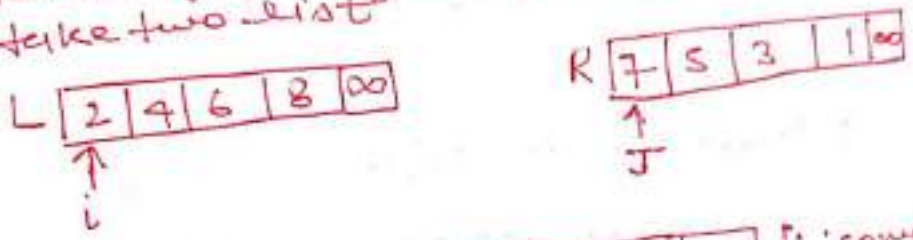
→ ~~Brief Explain~~

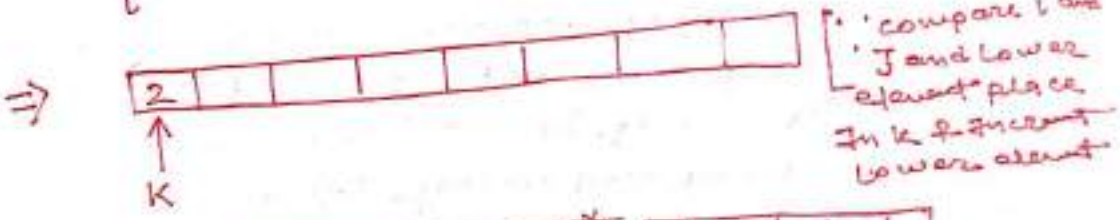
→ Let understand by Example.

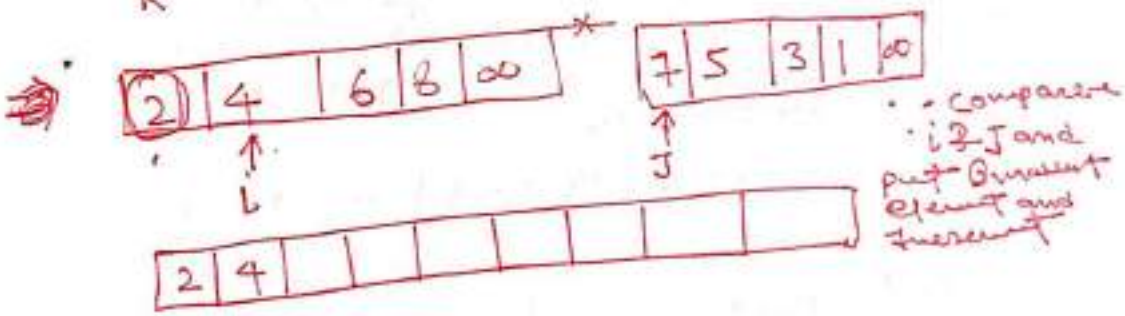
according to the Question, we have two list like that



According to the merge algorithm we take two list







follow this process until you find the entire sorted array

★ Second way to complete the question.

1- one given array is already sorted.

2- Second array is in descending order So apply Merge in second array and make it in ascending order.

3- Apply Merge in both array which are already sorted individually & compare till last element is not sorted.

★ Total time complexity

In given array we are compare two list. In given list total no. of elements are n. So we have to do n comparison and every time we compare the list (i & j) and written into new array means n elements copies into new array.

So we have $(n+n) = O(n)$

Total time complexity require $= O(n)$

Design a data representation which sequentially map n data objects into an array a[1,n], n1 of these data objects are stacks and remaining n2 equal to n-n1 are queues. Write algorithm to add and delete elements from these objects.

★ Insertion of elements in stack using array

```
Int Stack[max]
 Int ToP = -1
Void push (Int Item)
 {
 if(ToP==max-1)
  printf ("overflow");
  else
   {
   Top=Top+1;
   Stack[Top]=Item;
     }
    return;
      }
```

★ Deletion of element in Stack using array

```
Int pop ( )
 Int temp;
 {
 if (top==-1)
  {
 printf ("Underflow");
  return -1; }
   else
  { temp = Stack[top];
     top= top-1 ;
        }
     return temp;
       }
```

★ Insertion of element in Queue.

QInsert(Queue, N2, front, Rear, Item)

1- [Queue already filled?)
if front = 1 and Rear = N2 or if front = Rear + 1
then write overflow & return.

2- [Find New value of Rear)
if front = Null then (Queue initially empty)
Set front = 1 and Rear = 1
else if Rear = N2 then Set Rear = 1
else
Rear = Rear + 1.

3- Set Queue[Rear] = Item

4- Return.

★ Deletion of elements in Queue.

1- (Queue already empty)
if front = Null then
write underflow and return;

2- Set Item = Queue[front]

3- [Find New value of front]
if front = Rear then
Set front = Null and Rear = Null
else
if front = N then
set front = 1
else
set front = front + 1

4- Return.