MERGING Step 1: start 6/ep 2 : Declare the Variables Olep 3 : Read the 612e of the first array Otep 4: Read elements of first array in sorred order. Step 5 : Read the 612e of Second array 61ep 6: Read the elements of second array in souted arder. Otep7: Repeat otep 8 and 9 while i'm suco Step8: check IP a[i]>= b[a] then c[k++]=b[i++ Step 9 : else c[k++] = a[i++] Step 10: Repeat Step 11 while i'k mi Step 11: C[K++] = a[i++] 6-lep 12: Repeat 6-lep 13 while j'so Step 13: C[K++] = b[1++] Step 14 :- prolat the first array otep 15: proved the second array Step 16: polar the merged array Step 17 : End.

oulput Enten number of elements in array 1:4 Enter array 1 in sortled orders: 1, 3, 5,7 Enter number of elements in array 2: 4 Enter arrayin sorted order : 1, +, 6,7 first array is: 1,3, 5,7 second array 16: 2,4,6,7 meaged assay is: 1,2, 3, 4, 5,6,77

## STACK OPENTIONS

Step 1 : Hard step a: Declare the node and the required Variables olep 8 : Declare the functions for post, pop display and search an element slep4: Read the oboice from the used. Gleps: If the user choose to post an element then read the element to be pushed \$ call the function to push the element by tassing the value to the function Olep 5.1 : Declare the new node & allocate memory for the newnode dep 5.2: sel Newmode -) dala - value slep 5.3 : check IP-top == null -lben set Newmode -) next = null Steps. 4: Set newnode -> Next = top 6/eps. 5 : sel top = newnode &-then point insertion o is successfull 6tep 6: If user choose to pop an element -Prom the stack then call the function to pop the element dep 6.1: check IP top == NUII then provot dade is empty step 6.2: Elaxe declare a pointer variable temp and initiallize it to top step 6.3: point the element that being deleted Step 6.4: set temp = temp -) next step 6.5 :- Pree the temp

```
Step 7: If the usex choose the display theo
         Call the function to display the element
         in the Stack
Slep 7.1: check If top = Hull then provid stack is
          empty
Step 7.2: else declare a paroles variable -lemp
          & loitialize H to top
step 7.3 :- Repeat steps below while temp -> next
          1=001
Step 7-4: provot temp -) data
Step 7.5 : Set temp = temp - next
        :- IP-the user choose to search anelen-
Step 8
           end from the stack then call the
          -tonction to search an element
         :- Declare a pointer variable pts and
Step 8.1
           other necessary variable
otep 8.2: Initialize pto=top
61ep 8.8 : check if pto = null then provid stack
           empty
step 8.4 : else Read the element to be searched
          :- Repeat step 8.6 to 8.8 while ptil=null
6tep 8.5
          : check if pto- data == Item then
8-6 8-6
           priori element founded and to be loca-
           ted and sel flag=1
         : else sel flag = 0
Step 8.7
          :- locrement is by a and set prix = prix+)
Hep 8.8
         :- check IP flag=0 then provod the
 6lep 8.9
           element not found
          :- end.
 step 9
```

```
output
 choices:
1. push
2. pop
3. Display
4 Seanch
5. Exit
Edler Your choice : 1
Enler the value to be insert : 7
losertion is successfull
 1. push plans a gale broggs a Frank
choices:
2. pop 100d = CLEDD 91 stade & Boyala
 3. Display De Best Paris
 4 · Search all to the roll beneat to angul
 5. Exid THID THE TO SUITE
Enles your choice: 2
popped element: 14
choices property freely said houses & plants
 1. push booken auto later all gold
 8. Display
 4. search
 5. Exit
Enter your choice: 3
HOW (- F
Enteryour choice : 4
```

Earles the Hers to be seasched : 2 Hem not found choices: 1. pust 2. Pop a. Display 4. seasch 5. Exit Enter your choice: 5

## Ciacular Queue operation.

Step 1: start

step 2: Declare the queue and other Variable

step 3: Declare the function for enqueue, deque

step 4 :- Read the abova - from the usen

steps: If the user choose the choice enqueue then Read the element to be loserated from the user and call the enqueue fonction by paining the value

Step 5.1: check IP front == -1 \$\$ Rear == -1

then sel front =0, Rear = 0 and sel

queue [rear] = element.

Step 5.2: else IP Rear + 1 % max == front ox front = rear + 1 then print queve is overflow

Step 5.3: else set rear = rear +1 " max and set queue [rear] = element

step6: 1P-the user aborce is the option deque-

Step 6.1: check if front ==-1 and rear ==-1

then print queue is underflow

step 6.2: Else check IP front == Reas + then provided the element is to be deleted + then set - front = -1 and Reas = -1

6tep 6.3: else provol the element to be deque-

step 7 :- IP the user choice is to display the queue theo call the function display

step 7-1 : check if front = -1 and rear = -1 -1 bed
proved queue is empty

Step 7-2 : else repeat the step 7-3 while Pt seas step 7-3: provid queue [i] and set 1-141% max : If the user choose the search-then call Hep 8 -the function to search an element in the :- Read the element to be searched to Step 8.1 the queue : check IP Hem == queue [1] - then provat Step 8.2 Hen found and Hs position and locrement i by 1 :- check If c == 0 then provid Hem out Step 8.3 found :- end. Step 9

```
output
    choices :
1. loserd
     2. Delete
     a. Display
     4. Gearch
     6 Exit
    Exten your choice :- 1
    Enter the number to be inserted : 7
    choices in the second second second second
    1. lose of and a method and and
     2. belele
     3. Display
     4. search the other method as a sequilar
     5. Exit
          Palling a sta M Shoots a B & colo
    Enter your choice: 1
    Enter the number to be loserated: 14
   choices : A a gale lawrence
    1. Insert
             and of all deads
    2. Delele
     a Display
     4. Search
    5. Exit
    Enler your choice ?
     7 was deleted
    choices:
      1. losest
      2. Delete
      3. Display
```

Enter your choice: 3 14 choices : 1. logest a. belete 3. Display 4 beasch 5. Exid Earles your choice: 4 Earlex the element to be seasch: 14 Hem found at location: 2. A TOWNSON WORKERS IN FORMS met fall to the terrestal and the second of the second

## Doubly linked list operations. step1: start Step 2: Declare a structure and related Variables step 3 : Declare functions to create a mode loser a code in the beginning at the end and given position display-the list and search an element in the Ust step 4 : Define function to create a nade, declare the required variables step 4.1: Set memory allocated to the node = temp then set temp -) prev= null and temp - next = noll Step 4.2: Read the value to be losented to the oak step 4.3 : set temp -> n = data and increment Count by 1 Step 5 : Read the choice from the user to perform different operation on the list :- IP the user choose to perform losention operation at the beginning then call the function to perform the losertion Step 6.1: check if head == noll then call the function to create a node perform Step 4 to 4.3 step 6 2 - set bead = temp and temps = bead otep 6.3 = else call the function to create a node perform step 4 to step 4.3 to set top -> next = head, set head -> prev =temp and head =temp :- If the user choice is to perform Hep7 losextion at the end of the list then call the function to perform the losertion at the end

step 7.1: check IP head == null then call the function to create a new node a newnode then set temps head and then sel bead = temp1 Step 7.2: else call the function to create a new node then sel tempt -> next = temp, temp > prevetemp 1 and temp1 =temp :- If the user choose to perform losertion Step 8 in the list ad any position the call the function to perform the Insertion operation step 8.1: Declare a necessary variable step 8.2 : Read the position where the rade need to the losested, set temp 2 = head : check IP pos 21 on pos>= count+1 Step 8.3 then print the position is out of range : check If head == null and pos = 1 Step 8.4 then print sapty list cannot insert other than 1st position. :- check If head == null and pos=1 Step 8.5 then call the function to create New node then set temp = head and bead = temp1 step 8.6 :- while 92 pas then set temps = temps -) bext then increment 1 by 1 step 8.7: call the function to create a new node and then set temp -) preve tempa. temp -> next =tempa -> next -) prev = temp tempa -> next = temp step 9: If the user choose to perform dele tion operation is the list theo all the function to perform the deletion operation

```
Step 9.1 : Declare the necessary variables
step 9.2: Read the position obere node ned
           to be deleted set tempa = bead
Step 9.3 : check IP posks on post = court+1
           then print position and of Range
Step 9.4: check IP head == noll -then proval -the list
           15 empty
other 9.5: while Papas then tempa = tempa -) next
           and increment i by 1
Step 9.6: check If P== 1 then check IP-tempatinex
            + == pull then print node deleted free
           (Hempa) set tempa = head = null
Step 9.7: Check IP temp? -) next == null then
            temps -) prev -) next = null the o
            tree (-leopa) then print node deleted
Step 9.8: - tempa -> next -> prev = tempa -> prev ->
then check IP PI=1 then tempa -> prev ->
           next = temp2 - next
step 9.9: check If i== 1 then head =temp? ->
            next then print node deleted then
            free temps and decrement country
Step 10 : If the user choose to perform the
            display operation then call the function
            to display the list
step 10.1: set tempa= n
 Otep10.2: check IP temp2 = rull then proint
              list is empty
            :- while temp? -) next 1 = noll then pri-
 6kp10.3
              not temps to then temps = temps to next
            IP the user choose to perform the
 Step 11:
             search operation theo call the function
             to perform search operations
 Hep11.1: Declare the necessary variables
  step 11. 2 : step tepo = bead
```

Slep 11.3: check IP temps = noll then prior the last is empty Step 11. 4 : Read the value to be searched Step 11.5 : while temp 2 ! = noll the check IF tempa -> n= = data then prior element found at position count +1 Step 11.6: else sel tempa. tempa -> next and Increment count by 1 Step 11.7: prior element not found in the list step12 : god.

```
output
1 losed at begining
2. Insert at end
3. losered at position 1
4. Delete at i
5. Display from begining
6 Search for element
7 Exit
Erries choice 1
Enter the value of code : 3
Enler choice 1
Enter the value of nade : 4
Enter choice 5
linked list element from beginning: 4,3
Errier chaice :3
Enter the position to be inserted: 2
Enter value to node 1
Enter choice : 2
Enter value to pade: 6
Enter choice: 5
linked list element from begining: 4,1,3,6
 Enlex choice: 4
 Enter position to be deleted 1
  node deleted
 Enter choice: 5
 linked list element from beginning: 1,3,6
 Enter choice: 4
Enler position to be deleted: 3
 Node deleted from the list
 Errien choice: 5
```

loked list element from bogiolog: 1,3 Earles choice: 6 Enter value to be search: 7 Errox 7001 food in list Enter choice: 7 The person that the person have been been ministrate follows in release in the continues of the second ALCO 4.1.5 Sel warmer of the destroy of the seals has been a provide on a local contract the thoras later a family word out of single that the first of the land the state of the s at the same of the The state of the s the class to position the tolerand est bet at totals broods most thorn good to be of the series of a pole to be the control of the party and a ballion of a greet state of the were brown down board a female and the

```
Sel operations.
Step 1: Start
olepa: Declare the necessary variable
otep3: Read the choice from the user to perform
         set operation
Step 4 : If the user choose to perform Union
step 4.1: Read the candinality of a 2 sets
Step 4.2 :- check IP to 1 = n then priot cannot
          perform union
Step 4. 3: else read the elements in both the
step 4.4 :- Repeat the step 4.5 to 4.7 Ontil Pan
Step 4.5 - CEP = A [] B[]
Step 4.6 : print c[]
Step 4.7 :- Increment 1 by 1
Step 5 : Read the choice from the USEN to perfor
         am Intersection
Step 5.1: Read the condinality of 2 sets
Step 5.2: check IP mo|= n then print annual per-
          form Intersection
Step 5.3: else Read the elements in bath the sets
Hep 5.4: Repeat the Step 5.5 to 5.7 will Paro
blep 5.5 : CEJ = AED $ BED
Hep 5.6 : point CED
Step 5.7 : locrement 1 by 1
Step 6 :- If the user choose to perform det
           different operation
 Step 6.1: Read the cardinality of 2 sets
 Step 6.2: check If mi=n then print Connot perform
```

Step 6.3 :- else read the element in bob sets Step 6.4: Repeat the Step 6.5 to 6.8 contil iko Glep 6.5 = check IP A [] == 0 +ben c[] = 0 step 6.6: else IR BED == 1 then CED =0 Step 6 7 : else CLU=I step 6.8: Increment i by 1 Step 7: Repeat the Step 7.1 and 7.2 Until Pan Step 7.1 :- Print CED step 7.2 :- locrement i by 1

```
output
choice to perform
1 Union
  a. Intersection
a. Di-Pesence
  4. Ex?-
 choice : 1
 Earlex first set: 3
 Borles become sel: 3
 Earles firest set (0/1):1
 enter set (0/1)001
 Elements of set 1 Union set 2:101
 choice to perform
  1 copian
  2. loseration
  3. Difference
  4. Exit
  choice: 3
 Enler first set: 4
 Enter Second Set : 4
 Server first set : (0/1) 1
    0
 Enter second Set: (0/1)1
Elements of sel 1 - sel 2:000 1
```

choice to perform 1 conigo 2. Intexsection 3. Difference 4. Exit choice : 2 Enler first set: 3 Enter second sel: 3 Enter front set (011)1 Enter second set (011) lorexsection of sel 1 ... Intersection set 2 100 choice to perform 1. union 2. Insertion Cinter section) 3. Difference 4. Exit the court of the second choice: 4 Program Exit successfully. MAR MARIO But one

Binary	Search (	Tice

Step 1 - Start

Step ? Declare a structure and educture pointers for insertion eleletion and search operations and also declare a function for loader, traversal

Step 3. Declare a pointer as root and also the

step 4: Read the choice from the cises to perstain losextion, deletion, searching and located - traversal

step 5: IP the usex choose to perform losertion operation then send the value which is to be loserted to the tree from the user.

otep 61: pass the value to the lose of pointer and also the roof pointers

sleps-2: check if ! rood then allocate memory

steps 3: set the value to the lofo part of the soot and then set left and sight part of the soot to null and setus a soot

otep 5 4: check IP 2001 - Info - x then call the losest pointex to losest to left of the 2001

Glep 5.5: - check If 2001 -) lofo (x then call the losest pointex to losest to the right of of the 2001

61eps-6 - Return the Root

step 6: If the user choose to perform deletion operation then read the element to be deleted from the three partite root pointer and the Hem to the delete pointer.

```
otep 6:1: check If not pto then protol node not
 Step 6. 2: else If pla -> lofox x -the call delete
            pointer by passing the right pointer
            and the Hem
 step 63: else If pla -) lo fosx aben call delete
           pointer by pawing the left pointers
            and the Hem
 Step 6.4 + check IP pto -> 10fo == Hero then check
            If pto -) left == pto -> right then
            three pts and setus null
step 6.5 = Else IP pro-> left == null then set
        pt. pts -) left and free pts . xotusop)
Step 6.6: Else IP pta - right == null then set
            P1. pla -> sight and free pto setusna
Step 6.7: else set p1 = p1x -> right and p2=
           page cold.
Step 6.8 = while p1 -> left not equal to soll
             Bet py -> left pty -> left and free
             Ed arotas etd
Step 6.9 :- Return ptr
           If the user choose to perform
 Step7 :-
            search operation the call the pointer.
           to perform search operation
Step 7.1 :- 180+00 com Declare the necessary
           pointers and variables
 6tep 7.2: Read the element to be searched
 step 7.3 : obile pto check IP Hem> pto ) lofo
           theo pto=pto-soight
  Olep 7:4: Else IF Hero < p-18 -> lofo then
             pto=pto-) left
```

Step 7:5 : Else break step 7-6: check IF pto then provid that the element is found Step 7.7: else print element not found in tree and setusp soot step 8: If the user choose to perform traversal then call the traversal function god park the road pointers. step 8.1: IF 800+ not equals to not Recursively all the function by parsing soot sleft step 82: print root - 10fo Step 8.3: - call the 18 aversal function secursi very by passing root -) right

output I loser in binary free 2. belete from binary tree 3. Inorder - Iraversal of binary tree 4. Search 5 Exid Enter new element: 10 300 is 10 loorder traversal of binary-free is: 10 1. Insert in binary tree 2. belete from binary tree 3. Inorder traversal of binary tree 4 Seasch 5. Exit Enter choice 1 Enter new element: 12 800- is 10 Isorder traversal of binary tree: 10,12 1. Insert in binary there 2. belete from binary tree 3. Inorder traversal of binary tree 4. seasch 5 · Exit Enter choice: 1 Enter new element: 16 Roof is 10 loorder traversal of binary tree: 10, 12, 16

```
1. Insert in binary tree
   2. Delete from binary tree
    3. Inorder traversal of binary tree
    4. search
    5. Exil
  Enter choice: 1
 Errler new element: 14
  loorder traversal of binary tree: 10, 12, 14,14
   1. loserd to bloomy tree
   2. belete from binasy tree
   3. Inorder - Inaversal of binary tree
   4. Search
  5. Exit
Enter choice : 1
  Enter new element: 15
  800+ is 10
 loorder traversal of bloogy-tree: 10, 12, 14,15
 1. losered to binary tree
2. belete in binary tree
 3. loorder traversal of bloggy tree
4. Seasob
 5- Exit
Errier choice : 2
Enlex element to be deleted :- 15
10,12,14,16
1 loserd in binary tree
2. Delete from binary tree
3. Inorder traversal of binary-tree
```

Enter choice: 3 Inorder-Inaversal of binary tree is 10, 12, 14, 16 1. loser in binary tree 2. belete from binary tree 1 3. loorder traversal of binary tree 4. Seasch Erriex choice : 4 search operation in binary tree Enter the element to be seasched: 16 Element 16 which was beauched is found and 15-16 1. loseast in binary tree 2. Delele from binasy tree 3. Inorder - leaves al of binary lee 4. Search 5 Exit charce 5

```
Disjoin set [create, union, Find]
Slep 1: Hart
Slep 2: Read the number of elements from the
         user and store H in to the dis n
Hep 3: call function make selenthen
Hep 4: bet ?= 0
Steps: Repeate for P2 dis. n-then
      set dis parent [c] = i
          bet dis Rank [i] = 0
           Sel ?= ?+1
          Lend for loop?
Step 6 - user select the union operation then
Step 7 : Read the elements to perform union
        and store into x and y respectively
Step 8: perform find operation in with x
         set per-form step 2
step 9: If x set == y set then [endy]
         It dis Raok [x bet] < dis. Raok [4
Step 10 :
           sel) - then: sel dis parent [x se
           = y set set dis Rank [x 6et] =
            Lead It
Hep 17: else If dis. Rank [xsel] > dis . Rank
          [4 set] then
          SET dis. parent [4 set] = x set
           SET dis . Rack [x set] = dis Rack
                Txse++1
            SET dis Rank [4 sel] = -1
```

```
Step 14: It uses choose find operation then
6tep 15 : Read the elements to check It and
           store the value into the Variables
           x and y sespectively
Step 16: It find x = = find y then
Display "Connected Components"
6tep 17 :- else
            Display " No connected Components"
Step 18: If user select the display operation
          -Ben
6tep 19: Set 1=0
Hep 20 :- Repeate for Pk dis n toen
           print dis . parent [i]
           Let P= P+1
          [end of for loop]
Step 21: 1+ dos · parent [x] = x
         set dis · parent [x] = find (dis · par [x])
         setum dis pasent [x]
Step 22: Exit
```

```
Colpul
How many elements : 5
     menu
 1 00100
 2 Food
 3 Display
 Enter above : 1
 Enter the elements to perform union = 3
Doyou wish to Continue ? (1/0):1
         menu -
  1. 00100
  2. Food
  3. Display
 Enles choice: 3
 pakent array : 0 1224
 Sank assay : 00 1-10
 Doyou want to continue? (110)
       menu .
   1. 00100
   a. find
   3 Display
  enter charce: 2
Earlex the elements to check 1-1 Corrected
 Components
```

1-32 0101210 Connected Composents Parties & to could. Cont. I Coion o pormer sito bross and a find it sole has are 3 Display I am mellowet the sage 0=11-0 + 4000 Entler choice state a Reveale for Paris Land Enter the elements to check It Connected Companents 1 Connected Components Doyou aust to Continue ? (1/0) Send the electronis in Tother The lood to get a les x 91 & pole Alabaratha (to s) does to be a color topang 216 152 1 mode (150 one allow I have I show what had it is gold In los el dons are ros