Aim: Merge two Sorted arrays and store in a third Progro 1: Start Step1: Declare the variables. stopa: Read the size of first array step3: Read elements of first array in sorted order. Step 4: Read the size of second array. 8 top 5". Read the elements of second array in sorted order Step 6: Repeat step 8 and 9 while is and jan Step 7 Cheek ef a Ci] == bGj] then cOk++] = bGj++) step 8. Check of acij = bcj Then else cck+1]=a[i+1] step9: Repeat step 11 while 1km step 10. step u: c [k+1] = a [1++] Respeats 8 top 13 while jun stepla अक्रा । अ C[K++] = 6Cj++] step 14: Print the first array 8tph; Print the seand cirray 8tep 16 Priot merged array step 17.

Output

Enter assay 1 in scorted order: 2678

Enter assay 1 in scorted order: 2678

Enter size of array 2:3

Enter array 2 in sorted order: 135

Merged array 15 1235678

Airo: Singly linked Black - Push, Pop, lineau seasch. Stepl: Steat stipo: Declare the node and the required variables. step3: Declare the function for push, pop display & search. Read the choice from the user to push, pop display or search an element. step 5: If the ceses choose to push an element, then used read the element to be pushed and call the function to push the element by passing the value to the function. Step 5.1 Declare the new Mode and allocate memory for the Set new Node > data = value Check if top== null then set newNode-> next=null Else set new Mode > next = top Step 5.5 Set top= new Node and then print insertion is step to 1f the uses choose to pop an element from the stack than call the function to pop the element

step 6.1. If the Check of top== well then print stack is Step 6.2. Else declare a pointer variable temp and initialize
15 to top. step 6.3. Point the element that is being deleted. step 6.4. Set temp = temp -> next. step6.5. free the temp. step 7. If the uses choose to display the element in the stack then all the function to display the element in the stack. element in the stack. Step 71. Check if top == NUII then pant stack is empty.

step 72. Else declare a pointer versiable temp and

initialize it to top. step 7 3 Repeate step 74 and 7.5 while temp > neat!=null step 74. Point temp > date Sep 7.5 let temp = temp = next If the weser choose to search an element from the stack then call the function to search an element Step 8.1 Declare a pointes variable pts and other newssary variable.

step 8 a Initialize pts = top. sleps. 3 Check if pts = Null, then print stack is empty. step 8.4. Else read the dement to be searched from user. step 8.5 Repeat the step 8.6 to 8.8 while pts 1 = null. step 86 Check if ptr > data == item them point olement found and it's location and set flag = 1 Step 8.7. Else set flag =0 step 8.8 Invoement; by , and set pto=pto=next. Check if flag == 0 then print element not found-Step 8.9 Step 9: stop.

Output 1. Push 2. Pop 3. Display 4. Search 5. Exit Enter your choice:2 Stack is Empty. 1. Push a. Pop 3. Display 4 · Search 6- Enit Enter your choice: 3 Stack is Empty. 1. Push a. Pop 3. Display 4. Search 5- Eait Enter your choice: 1

Entes the element to be inscrted: 34 Inscrtan is success.

1. Push 2. Pop 3. Display

4. Search

5. Enet

Enter your choke : 4

Enter item to be searched: 56

Short - Paster For

france and clarent

the charact to be per

the less choose to push

Item not found.

1. Push

2. Pop

3. Display

4. Search

5 Enit

Enter your choice: 5

Div de

tops - will there see

Thomas &

type was Node and

pagno 3: Aino: Ciscular Queue step 1: Start seps: Declare the queue and required varechles. step3: Declare the functions for enqueue, display and search. Read the choose from the user to enqueue, dequeue display and seasch. step 5: If the uses choose the extrem enqueue then read the dement to be inserted from the uses, then eall the function enqueue and pass the value to the function.

step 5.1: Check if fount ==-1 and requ ==-1 then set front=0, reas=0 and set queue [reas] = element.

Step 5-2: Else 16 reast 1 mod mag == front or front ==

Reast 1, then point Queue is overflow.

Stop 5.3: Else set rear = rear+1 mod man and set queue

[reas] = element.

step 6: If the user choose the option dequeue then call the function dequeue. step 6.1 check if foint == -1 and rear == -1, then print queue le anforfloro. step 6:2. Else check of front == seas then point the element is to be delected. Then set front = -1 and rea =-1 Step 6:3. Else point the element to be dequeued. Set Scort = front+1 med man. step 1: If the uses choose the option to desplay the queue the call the functions deeplay. step 71 Check of front = 4-1 and reas ==-1 then print Queue is empty. Step 4.2. Else repeate the step 13 while it = sear. step 73 pamt queue CiJ and set 1= i+1 mod mans. step 8. If the user choose to seach an element in the queue, then call the function to search an element in queue. Step 8.1 Read the element to be searched in the queue.

Step82. Cheek if item==queue [i] then print item found and its position and increment c by 1. clip 8:3 Check if c==0 then print item not found. s. Insite at speafic because step9. stop. 4 Delete at speufic location 5. Deplay at specific bealing a search for clament t fact Enter choice: Extervalue la nate: d Edu choia. g Entre value la nocle: 5 Eda dais 3 Enter position to be mereted 4 Position and of some to inself Enter choia 4 Entre perhan to be deleted a alote delibert from list to winds where destal let elements from desputy 3

```
: Menu !!
1. Enqueue
2. Dequeul
3. Display
4. Search
Enter any option: 1
Enter a number to insert: 56
: ! Menu! !
1. Enqueue
2. Dequeue
3- Display
4. Search
5 Enit
Enter eine
           option: a
 56 was
          deleted.
" Menu!
```

1. Enqueue

2 Dequeue.

3- Display

4. Souch 5-Enit Enter any option: 3 The circular queue is emply! nothing to display. formings for many with make :: Menu! 1. Enqueue 2. Doqueur 3. Display all more people all pass 4 - Search 5- Eout Enter any option:

ent ent the element to be searched 43 term not found.

mile referred promp for from so

thought to the first war to the form of all

Holder II make Chen nest . 1 -+ 100

Pagna: Doubly linked List Declare a straicture and related structure variables. steps. Declare function to create a mode musest a node in the begining, insertion at the end, insertion at the given position, display the list and search in clement in the lest. Step 4: Define a fanction to create a node, declare the requered reartables. Step 4:1 Set memory allocated to the nede = tomp. then set temp > poer = null and temp > next = null. Step 4:0 Read the value to be inserted to the node. Step \$ 3. Set temp-> 1 = data and increment count by 1. 8 tep 5: Read the choise from the user to perform different operation on the list. 8to 6: If the uses choose to perform insertion operation at the beginning then call the function to perfeon

```
the insertion.
steps.1 (heck if head = = null then call the function to create a mode, perform step 4 to step 4 3.
 step 6.2 Set head = temp end lemp1 = head.
 step 63 Else call the function to execute a node, profession
       Step 4 to step 4:3. Then set temp > next = head
      set head -> prev = temp and head = temp
step 1: If the uses choose to perform insisters operateins
      at the end of the lest, then call the function
      to perform the insertion at the end.
Step 7.1 Cheek if head == null then call the function to
      create a new noele then set temp-head and then
      set head = tempt-
Step 12. Else all the function to create a neconode then
      set templ > next = temp, temp > prev - templ and
    temp1 = temp.
steps: If the user choos to perfero Prisestin operator
```

function to perform the inscolute operators Step 8.1 Declare the nocessary variables. step 82. Read the position wohere the noode mode med to be inserted, set temps. head. step 8:3: Check if pos x 1 os pos>= count +1 then print the step 8:4: Check if head == null and posi=1 then print "Empty list and cannot insert other than!" position. step 8.5. Check if head == nell and pos =1 then call the function to create new node, then set temp-head and head = tempt. tempa = tempa > next, then step 8:6 While ix pos then set inarment i by 1. step 8.7 Call the function to create a new node and then set temp> prev = tempa - temp> next = tempa > next temps > next -> prev= temp. tempa > neart = temp.

stop 9: If the cises choose to perform deletien operation to the list than call the function Step 9:1 Declare the necessary variables. Step 9:2 Read the position where node need to be deleted.

Set temps = head. Step 9's Cheek if pos 21 or pos> = count +1. then pent positions out of sange. step 9.4. Cheek if head == nell then peint the list is emply. step9 is while in pos. then lemp2 = temp2 > next and unrement ite Step 9.6 Cheek of i==1 thun cheek of fempe > nent == will then pent node deleted. free (temp2) set temp2 = hoad=null. Step9:7 Cheek if temps > nent == rull then temps > prw-nent-null then free Clemps) thun print node deleted. Step 9:8 temp 2 > next > prev = temp2 > prev, then check if it =1 then temps - prev - ment = temps - ment. Step 9.9. Check if i=-1 then head = temp2 > next this peint node deleted then free temps and

```
decrement count by 1.
 step 10. If the uses choose to perform the display operation
       then call the function to display the list.
App 10.1 Set tempa=h.
step 10 2. Cheek of ferrapa = null then punt last is emply.
dep 10.3 While temps - next 1 = null those point temps > n
those temps = temps = next.
step 11. If the word choose to perform the search
       operation then call the fensetien to proform search
step 11.1. Declare the necessary variations.
step 11-2 Set temps-head.
Step 11.3. Check if lemps == rull then point the set lest is empty.
 Step 11.4 Read the value to be seenched.
 Step 11.5 While tempa! = well the check of tempa>n == clate
 Step 11.6 Else set temps. temps -> next and increment a count by 1
 Sep 11.7 Paint element not found is the list.
```

1. Insest at begåning 2. Insut at end 3. Insut at specific location. 4. Delete at specific location 5. Display at specific localism 6. Search for element. 7. Exit. Enter choice:1 Exter value to node: 2 Enter choice: 2 Entre value to nocle: 5 Enta choice: 3 Eula position to be Prosected: 4 Position out of range to insut Entee choia: 4 Entre position to be deleted: 2 Mode deleted from list. Enter choice: 5 Linked list elements from begining: 3 Enter choice 6.

Dala found in 1 gosition Poully and to Enla choice: 7 (program finished) provide to mente a media present a media and the sall to estimate principal in the Since bearing by and the part and cleared to the list the state of contract of modern of anist it mining allocated to the mother temps there and two Main system is quelled in the test of the sales the the sales and the state of the second to the life chair from the cist

Pignos: Set Data Structure and Set Operations. Enter clements of record cets (0/1) 1 1 1 000: 6 30 - 355 do fordis Declare the necessary variable. steps. Read the choise from the uses to perform st operation step 4. If the user choose to perform union step 4.1 Read the cardenality of too sets. dep 4.2 Cheek if m1=n then peint cannot perform amon step 4.3 Else read the elements in both the sets. step 4.4 Repeat the step 4.5 to 4.7 until 1km Step 4:5 (Ci] = A Ci] BCi] Step 4.6 punt c Ci] Step 4.4 Increment P by 1 perform Prestruction. step 50 Read the choise from the used to steps. 1 Read the cardinality of two sets. stepso. Else read to elements in both the sets. Step 5 2 Eta Check if in != n then print cannot perform instruction

```
step 5-3 Hse read the elements in both the sets-
step 5.4 Repeat the step 5.5 to 5.7 custoliam.
dep 5.5 CCIJ = ACIJ | BCIJ
step 5 6 print (Ci)
step 5.7 Increment 1 by 1
steps. If the user choose to perfer set difference operation
Step 6:1 Read the cardinality of two sets.
step 62 Cheek if not = 10 then print cannot perfer set
   different operation
step 6.3 Else read the elements in both sets
step6.4 Repeat the step 6.5 to 6.8 until 120.
step 6.5 Cheek if A [i] = -0 then C[i]=0
Step 6.6 Else if E[i]==1 then C[i]=0
Step 6= 1 Else Cli]=1
Step 6.8 Increment i by 1
Step 7 Repeat the step 7-19 7-2 until 12m
step 7.1 peut C [i]
 step 7 a increment i by 1
```

Output 1. Input choice to perform; 4. Exit. 1. Union 2. Intersection 3. Difference Enter cardinality of first set: 4 Enter cardinality of second set: 4 Entre elements of first 2t: (0/1)1001 Enler elements of second set: (0/1)0110 Elements of set 1 union set a: 1111 Input choice to perform: 1. Union a Intersection 3. Difference 4. Enit Choice: 2 Enter cardinality of first set? 3 Enter cardinality of second set: 3 Enter elements of first set: (0/1) 110 Enter elements of second set: (0/1)100 Exements of sets intersection set 2: 100 Input choice to peeferm. 1. Union a. Intersection 8. Difference 4. Enit cardinality of first of

Enter cardinality of second set. Enter elements of first set: (0/1) 101 Enter elements of second set: (0/1) 1 1 1 Element of set-set a :000 dans pristage sell conso والم مدور والمدهد العراد the test the contending of the sets of The town of the part of soil of the 100 mm 10 mm A Liberton 1-100 mil part the chara gran the and the state of the sale of the sale

Aim: Pomagy Sarch trees. steps Declare a structure and structure pointers for msention deletion and search operations and also declare a function for inorder traversal. Step 3: Declare a pointer as root and also the required Step 4: Read the choise from the user to perform insurtion, deletion, searching and morder traversal. steps: of the user choose to perferm inscrition operations then read the value which is to be insisted to the tree from the uses. step 5.1 Pais the value to the insert pointer and also the root pointes. for the roof. Step 5.2 Check if 1.100t there allocate memory the root step 5.3 Set the value to the info part of the soot and then set left and right part of to well and return roof-

- step 5.4 Check if 200t -> info-> x then call the insert pointed to insert to left of the 2001.
- step 5.5. Check if root > info < a then call the insert pointed to insert to the right of the root.
- step 5.6. Return the root
- step & If the user choose to perform deletern operation than read the element to be deleted from the tree.

 Paux the root pointer and item to the delete pointer.
- Step 6.1 Cheek if not pto then print node not found.
- Step82 Else if pto sinfoxa the call delete pointer by paring the right pointer and the item.
- step 63. Else if ptr > info>10 then call delete ponter by passing the left pomlés and the item.
- Step 6.4. Cheek if ptr > info==item than cheek if ptr > left==

 ptr -> right then free ptr and relians rull.
- step 6.5 else if pto > left == nell then set PI. pto > right and free pto, setuen PI.
- dep66. Else if pto = right = well then set P1 = pto > left and free, pto, return P1.

Step 6-7 Else Set PI = pto > right and Pa a pto > right. dep6 8 While PI -> left not aqual to well, set PI -> left = pto >'left and free pto; return P2, step 6.9 Return Pto Step = If the uses choose to gentern search operation the call then pointer to perferm search operation. Step 71 Declare the necessary pointers and variables step 12 Read the element to be siarched. dep 13 while pto cheek if item> ptr > info thon pto=pto = ight. Step 7 4 Else if item 2 pto > info them pto = pto > left step 75 the break. Step 76 Cheek if ptr then point that the element is found. Step 17 Else print element not found in tree and returns step 8. If the uses choose to perform traversal then call the braversal function and pun the roof pointer Step 8:1 If noot not equal to mill remesively call the function by pooming root > left. steps a Paint root -> into step 8:3 Call the traversal function recurrely by passing east > right.

Output I Insert to bisary tree 2. Delete from brinary tree 3. Inorder traversal of bornary tree. Scarch Enter chora: 1 Enter new element: 65 root is 65 trooder traversal of bornary tree is: 65 I must no binary tree a Insute from minary tree 3. Inorder traversal of bornag tree. Enter choice :1 Ente new elements: 55 Incorder traversal of language tree is: 55 65 1. Insect in bring tree a. Delete from bornary toce 3. Inosque traversal of

4. Seasch enter choice: 1 Enter neco elements: 46 Inordes baversal of brinary tree is 46 55 65 1. Inset in binary tree 2 Delete from binary tree 3. Inorder traversal of bomany tree 4 - Search

5- Enit

Enter choice: 4

Search operation in bornary tree Enter the element to be searched: 46 Element 48 which was searched is found: pregnot: Arm: Disjoint sets and Associated operations steps: Start
steps: Declare the structure and related structure variable step 3. Declare a function make set () slep 3.1 Repeat slep 3.2 to 3.4 antit is step 3.2 des parent [i] is set to i Step 8:3 Set dis. Rank (1) is equal to 6. Step 8.4 Inexement i by 1 step 4 Declare a function display set step 4.1 Repeat step 4.2 and 4.3 until 120 Step 4.2. Print dis parent [i] Step 4.3 Increment i by 1 4.6 until 120 Step 44 Repeat step 4.5 and Step 4.5 peint dis rank [1] Step 46 Increment i by 1 step 5. Declare a function find and pess x to the function dep 5.1 Check if dis parent Gil!= x then set the return value to dis parent [2]

```
dep 5 2 return dis parent [7]
 step 6 Declare a function union and pass too variables
 slep 6.1 Set x set to find (n)
Step 6 a set y set to fund (y)
Step 6.3 Cheek if x set == Y set then return
Step 6.4 Cheek of dis.rank [a set] & dis. sank [y set] then
step 65 st yest = dis parent [y set]
Step 6.6 Set 1 to dis rank [x set]
step 67 Else if cheek dis rank [x set] > dis eart [yset]
step 6.8 set x set to clisapment [y set]
steps 9 &f -1 to dis rank [y set]
step 6 10 Else dis. parent [g set] = x set.
Step G11 Set des. rank (n set] +1 to dis rank [x set]
Step 6-12 Set + to dos nank [y set]
step 1 Read the number of elements
dep 8. Call the function make set.
step9. Read the choise from usu to perform union
```

Find and display operation If the user choose to perfern union operation step10: reed the dement to perform anon operation. If the user choose to perform final operation read Step 11: the element to cheek if connected. step 11.1 Cheek if find (x) == find (y) then print connected component. Else print Not connected component Step 12 If the usu choose to perform display operations call the function desplay set.

Stop Step 13:

```
Enter the no: of elements: 7
   MENU
** * * * *
 1. Union
 2. Find
 3. Display
 Enter choice ! 1
Enter elements to perfern anion: 3
Do you wish to continue? (1/0)
  MENO
 1. Unlon
 d. Find
 3. Dusplay
Enter choice: 2
Enter elements to check if connected components:3
Connected components
Do you wish to contince? (1/0)
```

MENU: X ** ** * Y 1. Union a. Find 3. Display 14x Ame and my Enter choice: 3 Parent Assag. 012 3 4 3 6 Rank anay: 00010-10 00 lango d (1) dim . 25

Do you wish to continue? (16)

الرامي م المسدلدي

pow & H pole Ing.

and them at the entire

Red 1 Johnston

(Program finished)