MINTHO program no:-1 Aim: - merge two sorted arrays and store in a third array - 1 1 total Latina or 1 por a rather Algorithm Step-1:- Starts / who later of a part of part step-2: Declare the variables of the porter terril step-3: - Read the size of first array Step-4: Read elements of first array in sorted order step.5: Read the size of second array Step-6:- Read the elements of second array in sortted order Step-7: Repeat Step 8 and 9 while ixm and ixn step-8:- Check if a[i]>=b[i] then C[K++]=b[+++] Step-9: else c[k++] = a[i++] Step-10: Repeat Steps 11 while itm Step-11: C[K++] = a[i++] Step-12:- Repeat Step 13 while Ixn Stcp-13:- C[K++] = p[]++] Step-14:- print the first array

Step-15: - print the second d=10911 Step-17: - End. Fibationi ad ot applier Plitononala Laggad dang.1

1-1000 cmes Output:enter number of elements in array 1:4 enter array 1 in sorted order: 1357 enter number of elements in arrays: 4 enter array a in sorted order : 1 4 6 7 1 First array is : 13 52/4 pirer att areland is second array is: 24 6 of see out book is merged array is: 12345677 bood porto bosse to seis bot boss of see Peed the elements of second and the est too mai stide p bood 8 gots toogod for 198:61210 000+ (i)d= < (i)0 +1 1100 0 188 (1+1) = = [++>] > == 1= P9 cost states in sate hanged late (1+1) = (++>1) > = (+++) and allows El gold tongod - 61-7 [+++] d = [+++] > 111 Borra tert adt foing was

Program no:-2. 9009.8 Aim :- Stack operations Algorithm Step-1:- Start Step-2: Declare the node and the required variables Step-3: - Declare the functions for push, pop, display and search an element. Step-4:- Read the choice from the user. step-5:- If the user choose to push an element, then 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 node and allocate memory for the newrode. Step-5. 2: - Set new node -> data = value. Step-5-3: - check if top == null then set newNode + next dengy = null 909.0 Step-5.4: Set newNode > excht = top step-5.5: set top = new Node of and then print insertion is success ful.

- step-6:- if user choose to pop an element from the stack then call the function to pop the element.
- step-6.1:- Check if top == null then print stack is empty.
- step-6. D:- else declare a pointer variable temp and initialize it to top.
- Step 6.3: print the element that being deleted.
- Step-6.4: set temp = temp + nesct.
- Step-6.5: Free the temp
- Step-7: if the user choose the display then call the function to display the element in the stack.
- Step-7.1: Check if top== null then print stack is empty
- Step-7.0:- else declare a pointer variable temp and initialize it to top.
- Step-7.3: Repeat steps below while temp > next=null
- Step-7.4: print temp > data.

Step-7.5: - Set temp = temp > next. step-8: - if the user choose to search an eleme nt from the stack then call the funct ion to search an element. Step-8.1: - Declare a pointer variable ptr and other necessary variable. Step-8. D: Initialize ptr = top Step-8-3:- Check if ptr = null then print stack emp step-8.4: else read the element to be searched. Step-8.5:- Repeat step 8.6 to 8.8 while tri=null Step-8.6:- Check if ptr > data == item then print element founded and to be located and Set flag = 1. Step-8.7: else set flag=0 Step-8.8: - Increment i by 1 and set ptr = ptr + next Etep-8.9:- Check if flag=0 then point the element not found. Step-q:- End.

```
Emito beasse out tong
output:
                   worre toprom out tring on
choices
                                      baj ma
1. push
2. POP
3. Display
4. Search
5. exit
enter your choice :1
enter the value to be inserted: 7
Inscrition is successfull
Choices:
1. push
 2. Pop
3 Display
4. Scarch
5. excit
enter your choice: 2
popped element: 14
choices
1. push
```

D. push 3. POP 4. Display troto - 1-5. cocit enter your choice: 3 and show soft stabood A Null dornos bon Choices sale bead the choice train the citer 1. push 2. POP 3. Display no daug of soods took to at 71.30 ad of terments and book andt call the two choo to push the parsing the volue to the paisted 5. escit enter your olichoice: aporusa sat explasa - 18 tol the newnode. ented the item to be searched: 2 Hem not found ve ptobe & spoonwar too de les choices Prosest check it topes and then set neurlos 1. push Iliza 6 2. POP got these stollware too it 2.91 3. Display 4. search wing and two to about and got enter your choice: 5

Program no:-3 Aim: Circular Queue - Add, Delete, Search. Algorithm Step-1:- Statt Step-D: Declare a queue and other variables. Step-3: Declare the functions for enqueue, deavueue search and display. step-4: Read the Choice from the user. Step-5: if the user Choose the choice enqueue. from the user and call the enqueue function by passing the value.

then read the element to be inserted

Step-5-1: Check if front == -1 and rear == 1 then set front = 0, rear = 0 and set queue [rear] = clement

Step-5. D:- else if rearth 10 masc == front or front = reart then point queue is overflow.

Step-5.3: else set rear = reart 1 . I mase and set queue [rear] = element

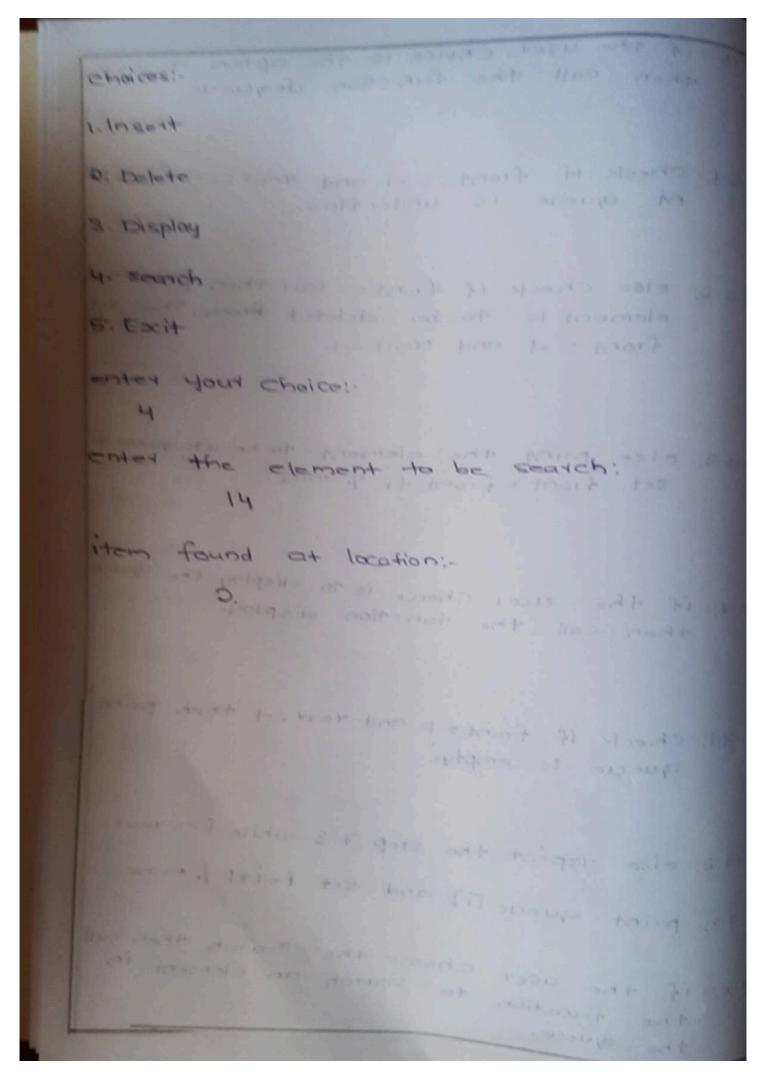
Step-6:- if the user choice is the option dequeue then call the function dequeue. Step-6.1:- Check if front == -1 and rear ==-1 then pi nt queue is under flow. volgoict is Step-6. D:- else check if front = = rear then print the element is to be deleted formand. Then set front = -1 and rear = -1. Step-63: else print the element to be dequeued set front = front +1 1. max. Step-7: if the user Choice is to display the queue then call the function display. Step-7.1: check if front =- 1 and rear =- 1 then print queue is empty. Step-7. D: else repeat the step 7.3 while it = rear. Step-7.3: - print queue [i] and set i-i+1:/, masc. Step-8:- if the user choose the search then call the function to search an element in

the queue.

Step-8.1: - Read the element to be searched in the que uc. Step-8.2: Check if item == quece [i] then print found and its position and incre by 1. Step-8.3: - Check if C == 0 then print item not foand. peringed disile pelged to Step-q :- End E H : pringed work how as to il hadril a: How of sulpp rates

-trans-gart gart output:-10210 - str +1 . 44 choices: The se of the second 1. In seit has and whole out mort to months and darper of coi 2. Delete 3. Display 4. Search in aldomov totolog 5. Excit Enter your choice: got and application is any 1 enter the number to be inserted up 7 Choices:- ad of tosonals and book sale it 8-1 1. Insert olida 8.8 of 0.8 gots toagod . 68.8 2. Delete 3. Display and moti = ptob ent 71 alondo - 23.0 top between to be boated and 4. Search 5. Excit Enter your Choice: 15 + 17 : 117 too has 16 pd 1 momoral -: 3.84 Enter the number not be inserted: sloods - part · forog. 100 .bn3

choices: I Insett Autor Holod Bla - Stone P tologels 2. Delete 3 - Display 4. Search 5. Exit to memors for goodfront and araland enter your choice: Bend the choice them the man 7 was deleted at month took and the then yead the relement Choices .. 1100 boo 1021 and mort l'Insert solor ent points points de noitsout 2. Delete 3. Display 1 small book to the thort the stand (car) some to bot of there of toot -knomsto 4. Search 5. Escit enter your choice: separa of 1+ most fineste setten 3 di surua triog and think enter the element to be inserted:



Program nos-4 Aim : Doubly linked list - Insertion, Deletion, Search, Algorithm Step-1:- Start Step-8:- Declare a structure and related variables dotal state step-3: Declare functions to Create a node, insert a node in the begining at the end and given position, display the list and Search lan element in the list Handrood a markers Step-4: Define function to create a node, declare the required variables. Step-4.1: set memory allocated to the node: temp. Then set temp > prev= null and temp> Desct = null. Step-4. 2: Read the value to be inserted to the node. 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. E: 331003 13103

- operation at the begining then call the function, to perform the insertion.
- ion to create a node, perform Step 4
 to 4.3.
- Step-6. 2: set head = temp and templ = head
- Step-6.3:- else call the function to create a node,

 perform Step 4 to 4.3 then set temps

 next=head, set head > prev=temp and

 head = temp.
- Step-7: if the user choice is to perform insertion at the end of the list, then call the function to perform the insertion at the end.
- Step-7.1: Check if head == null then call the function to create a new node then set temp = head and then set head = temp!
- step-7.2: else call the function to create a new node then set temp > next = temp.

 temp > prev = temp1 and temp1= temp.

- in the list at any position then call the function to perform the insertion operation.
- step-8.1: Declare the necessary variable.
- step-8.0: Read the position where the node need to the inserted , set tempa = head.
- Step-8.3: Check if pos < 1 or pos> = Count +1 then print the position is out of range.
- step-8.4: Check if head == null and pos=1 then
 print empty list Cannot insert other
 than 1st position.
- Step-8.5: Check if head == null and pos=1 then call the function to create newnode. Then set temp = head and head = temp 1.
- Step-8.6: while ix pos then set tempo=tempo>
 neset then increment i by 1.
- Step-8.7:- call the function to create a new node and then set temp > prev = temp 0 . temp > next = temp 0 + next > prev = temp. temp >> next = temp.

- step-9:- if the user choose to perform deletion operation is the list then all the function to perform the deletion operation.
- step-9.1: Declare the necessary variables.
- step-9.2: Read the position where node need to be deleted Set temp? = head.
- Step-9.3: Click if pos <1 or pos> = count+1 . then
 print position out of range.
- Step-9.4: Check if head == null then print the list is empty.
- Step-9.5:- while i <pos then temp 2 = temp 2 → nesct and increment i by 1.
- step-9.6: Check if i == -1 then Check if temp? >

 nesct == null then print node deleted

 free (temps) set temps = head=null.
- Step-9.7: check if temp 0 > neact=null then temp 0 >
 prev > neact = null then free (temp 0) then
 print node deleted.

Step- 9.8: tempo -> next -> prev = temp o -> prev then check if is then tempo sprey-snext =tempo =next. step-9.9: - Check if i=1 then head = tempo - next then print node deleted then free temps and decrement count by 1. 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 Step-10-2: Check if temp2=null then print list is compty. Step-10.3: while temp 2 -> next = null then point temps > n then tempo = tempo -> neact. Step-11: if the user choose to perform the search operation then call the function to perform search operations. Step-11.1: Declare the necessary variables

", hagtwo Step- 11. 2: - Bet temp 2 = head Step- 11-3: - Check if temp? == null then print the list is empty. points & Cotton 2 hatel G step-11.4: Read the value to be searched. Step-11.5: - while tempo! = null then check if tempo >n == data then print clement found at position count +1. 5 too taxit vators step-11.6: else set temp a = temp a - next and incre
ment count by 1 Step-11-7: print element not found in the list Step-10:- ERd! & too acion 1 too to stangels more to person For it maskified & activation a actival H tall brown 1 (110) the thirt rates

output:- letament and att transmale and book alla 1. Insert at begining war check if there = que til tem pros 2. Insert at end 3. Insent at position! 4. Delete at i torng and to == 2 71 31 3000 -18 -18 5. Display from begining 6. sourch for element 7. exit enter Choice: 1 enter value to node: 3 enter choice:- 1 enter value to node:-4 enter choice : 5 linked list element from begining: 43 enter choice : 3 enter the position to be inserted: 2 enter value to mode:- 1 enter choice: - 2 enter value to node : 6

linked list elements from begining: 4136 court tree enter choice: 4 enter position to be deleted: 1 bomber has protopres a prolong Node deleted. other of short of economit explant in enter choice: 5 to portaged wat of shoot given per perguit and property linked list elements from begining: 136 enter choice: 4 enter position to be deleted: 3 soited node deleted from the list enter Choice: 5 between promon too sinking linked list element from begining: 13 TIMO = 70000 cotes choice: 6 enter value to be search: 3 the state when the person is the Data found in a Position terson arrai bons stob and good tos at pas enter Choice: 6 enter value to be search: -7 error 7 pot found in dist

היסקימה חסי- 5 Aim: - set operations Algorithm Step-1:- start step-2: Declare the necessary variable. construct of mind step-3: Read the choice from the user to perform the user to perfo 6-1000000 step-4:- if the ciser choose to perform union. step-4.1: - Read the cardinality of 2 sets. 1 (110) too tooit rotes step-4.2: Check if m1=0 then print cannot perform union. Step-4. 3: - else read the elements in both the sets. Step-4.4: Repeat the Step 4.5 to 4.7 contil ixm. Stop-4.5: C[i] = A[i] | B[i] | +02 | 0 +00 ool a too northone ratel Step-4.6: - print c[i] court of at a room Step-4.7% Increment is by I not make that a moral Step-5: Read the Choice from the User to perf orm inter section.

Step-5.1: - Read the cardinality of 2 sets. Step-5.2: Check if mi=n then print connat perfointersection Step-5.3: else read the elements is both the sets. Step-5.4: Repeat the Step 5.5 to 5.1 until icm. Step-55: C[i] = A[i] & B[i] Step-5.6: print c [i] Step-5.7: increment i by1 Step-6: if the user choose to perform set diffevence operation. Step-61: Read the cardinality of 2 sets. Step-6. 2: - Check if mi=n then print cannot perf orm set difference operation. Step-6.3: else read the element in both sets. Step-6. 4: Pepeat the Step 6.5 to 6.8 until icn. Step-6.5: Check if A[i]==0 then c[i]=0

Stop-6.6:- else if B[i] := 1 then c[i]=0 as in provided of freed 1 Step-6.7: else C [i]=1 step-6.8: Increment i by 1 Step-7: Repeat the Step 71 and 7.0 until ixm Step-7-1: print C[i] Step-7.2: increment i by 1. sort propriet my tissult GI OF sort provid po

```
busid : a great tol
output:
choice to perform
                        ptomo al tell
1. union
2. Inter section
3. Difference of sold able sold and has a part
4. cocit
    I de souto mode mon e la genste olidus
choice:- 1 -- 10 taing and pholo sa com
              at position count to
enter first set: 3
enter second set: 3
enter first set (0/1):10 toward toward
                   cost a Gampt to 2219 134
enter set: (0/1) 0 01 tosonds toing Fill
elements of set 1 union set a: 101
Choice to perform:
1. union 2. Intersection 3. Difference 4. exit
Choice: 3
enter first set: 4
enter second Set: 4
enter first set: (011) 1
0
1
```

enter second Set: (0/1) 1 0 clements of set 1: - set a: 0001 ordered the second set propose choice to perform pred the choice from the city to fait 1. Union 2. Intersection 3. Difference 4.0 Choice :- 2 Enter Aist set: 3 enter second set: 3 philippin and borg . 1 enter first set: (011) 1 enter second set: (0/1) 1. Repeat the step 4.5 to 4.4 costs clements of set: 1 [1]0 | [1]0 : [1]0 Inter section set 2 100 [i] o toing Choice to pertorm 1. union 2. Inter section 13. Difference 40 choice: 4 1) with mort sames and prost program escit successfully.

program no 1-6 Aim: Binary Search tree to be to the laboration Algorithm Step-1: Start step-D: Declare a structure and structure point ers for insertion, deletion and search operations and also declare a function for inorder traversal. step-3: Declare a pointer as root and also the required variables. work provid 196 post bount Step-4: Read the Choice from the user to perfor ym insertion, deletion and searching and in order traversal. Step-5:- if the User choose to perform insertion operation then read the value which is to be inserted to the tree from the User. work proceed as product Step-5.1: pass the value to the insert pointer and also the root pointer.

step-5. D:- check if not root then allocate memory

step-5.3: - Set the value to the info part of the root and then set left and right part of the root to null and return root.

step-5.4:- Check if root > info > >c then call the insert pointer to insert to left of the

Step-5.5:- Check if root + info < ac then call the insert pointer to insert to right of the

sort provid out stated a

Step- 5.6: - return the root les levert inbient

Step-6:- if the user choose to perform deletion operation then read the element to be deleted from the tree pass the root deleted from the item to the delete pointer and the item to the delete

Step-6.1:- Check if not ptr then print node not fou

step-6.2: else if ptr > info < oc, then call delete pointer by passing the right pointer and the item.

step-6.3: else if ptr -> info >>c then call delete pointer by passing the left pointer and the Item.

step-6.4: Check if ptr > info == item then Check if

ptr > left == ptr > right then free ptr

and return null.

Step-6.5:- else if ptr > left == null then set pleptr > right and free ptr, return pl.

Step-6.6:-else if ptr > right == nall then set pl= ptr > left and pro free ptr, return pl.

Step-6.7: else set pi= ptr > right and pz= ptr> right.

Step-6.8: while PI > left not equal to null, set PI>

Left, ptr > left and free ptr, return
PD.

step-6.9:- return ptr.

Step-Edo: if the user choose to perform search operation then call the pointer to perform search operation.

Step-7-1: - Declare the necessary pointers and variables! Step-7-2:- Read the element to be searched step-7-3: - while ptr check if item> ptr ->info then ptr = ptr > right. Step-7.4: else if item < ptr > info, then ptr = ptr > left Step-7.5: - else break Step-76:- Check if ptv then print that the element is found. Step-7.7: clse print element not found in tree and return root. Step-8:- if the User Choose to perform traversal then call the traversal function and pass the root pointers. Step-8.1:- if root not equals to null recursively call the functions by passing root > left. Step-8. D:- point root +info Step-8-3:- call the traversal function recursively by passing root + right.

output: 1. Insert in Binary tree 2. Delete from binary tree 3. Inorder traversal of binary tree there will be the gote out tought 4. Search (i) 3 tains 14 s. exit enter choice: - 1 enter new element:-10 1 i tomorral 10 1 400+ is 10. Inorder traversal of binary tree is: 10 1. Insert in binary tree 2. Delete from binary tree 3. Inorder traversal of binary tree 4. Search 5. exit enter choice: 1 enter new element: 12 100+ Is 10 morded traversal of binary tree: 10 12 ilmsert in binary tree

2. belete from binary tree 3. Inorder traversal of binary tree 4. Search 5 · escit enters choice: - las simplimes a probad enter new element: 16 bas anoito ragio root is 10 that the rebreci tot morder traversal of binary tree: 10 12 16 1. Insert in binary tree string a prolond 2. Delete from binary tree beringst 3. Inorder traversal of binary tree. the Choice from the deed 4. Search & boos noite isto northezeri cor los toport tabroni bos 5. exit Cotter Choice :1 enter new clement: 140000 1920 ant 71. al border traversal binary tree: 10 12 1. Insert in binary tree 2. Delete from binary tree 3. Inorder traversal of binary tree 4. search 5. escit

enter Choice: 0 m number of our your out rat enter the element to be deleted: 15 10 12 14 16 Intert in binary tree about soft to 2. Delete bingry tree of took off to 3. Inorder traversal of binary tree 4. search most according toor 71 stood of to that of march of rataing tra enter Choice: - 3 Inorder traversal of binary tree is: - 1012 I Insert in binary tree 2. Delete from binary tree 3. Inorder traversal of binary free 4. search motored of second vesto ent files all adt been nedt come rado deleted from the street potalate enter choice: 4 more sale book ratalog search operation in binary tree throng enter the element to be searched: 16 dement 16 which was searched is found 15-16 1. Insert in Linary tree this ity fi sals -00 2. Delete from binary tree

3. Inorder traversal of binary tree ming that art possess to intolog 4. search and the land, while the 5. exit Senter choice: -5: afair Ha 71 >132000 1 and not onal ideas - rig = +771 < rig stant prospers of the sale sale in anter rig soft for tagir c 13 12 cont 1100 = 3 tdpir 4 +19 +1 0310 120 of maken , my want no boo that e rig 1. 6120 264 bis bit + 118 pt 209 bis 2019 -: Fre 8 19 point plant toot to still plant of the left, pro production track pro, track 09 Amus meeting of second rocks, out 71 of interior the los and contrargo portorno de de peration