;60 30 all and the second s UNIT-3 : sildua Lioned Lists \* A linked life is a collection of nodes (or) collection : (Joke biov 0, inter-connection nodes : ( )holdeib bion \* The linked list is a linear data structure in which the nodes are executed sequencially by the establishing connect from one node to another node. \* Basically the linked lists are classified into 3 types they are i) single linked list cout <<"enter 6 volue" ii) Doubly linked list (d << 0i) iii) circulage linked list. Single linked list:-Ubbo 118 biov In the single linked list each node contains two fields they are 1) that tields 2) link (or) next field Node structure: in the sinde linked listing wid Biss Data d-D=D Link (05) fed next field. Data field: The data field can be used to store the abta (or) an element. soles addition ">>>ta Link (a) next field the link field can be used to store the address of next node in the single linked list \*In the single linked list the link field of last node is NULL. : Ido 8 Gr 10, 20, 30, 40 il rozal obl-getdata to NULL bo . NO 2000 1500 1 30 20 10 1000 2000 due .100 1500 500 1000 obly displayers: "etwan o; (1d)+9B

approxitions of single linear lists :-On the single linked lists we perform four basic operations they are 1) create 2) Insert 3) Delete 4) Display Create in the create operation is used to create a node in oxdes to -fam a single linked list. Ingest; - This operation is used to insert a node cither front (or) middle car) lost to the existing node (3) in the single linked lift. Reletei- In this operation can be used to delete a specific node in the single linked list Display's this operation can be used to display a the liet of elements in the single linked list. Ezi- program to implement single linead list. # include < icetscarr, h> # include <conio.h> # Include < process.h> class slink ł antis salados Stouct node structure name of fred ( Fulliedd int data; node \*next; } \* head; public; for Con soal \$ Slink() default constructor contread = NULL for att of the other of UUNITE book inst node at nidle"; :(1 << m)

ा मेर्ग भाषा भाषा हिंदि । method functions das's board spore all m void create (); Void stoss? (1 and patt croiling, insert(); void delmade(); Noid displaye ) ; head is 1; Void slink: (creater) 10 NULL Coecilein the create of 2001 ich ک realer head = new node; abo Cast<<" Enter the data for the node;"; Cin>>head > chata; at have a nothing of intering at hort (53) metile (33) list is the existing meta) in the inde linual lit. Void Slink :: insert() tothomie at base at nos mare souls at protection chore p; tot based of apriliant flocof be billed note \*list \* poer, \*temp, as straitpan and proget -the ist at dements with the fight shall be the caut << "Insert the node at -bant/midule/last?f/m/1:" cin>>p; f (b==,t,) # for tops < press ł temp= new node; cartac" enter the data for the node:"; "how buck cin >>tomp ~>data; temp ->next =head; , stab thi iter xirest: head = temp; + \* head ; 3 : >1609 if(p=='m')2 cout <<" Enter the abita for the node before to insert re node at middle"; Cin >> n;Sec. 1

while (list  $\rightarrow dital=n$ ) there boost - boost Poer = lift; list = list ->next; temp = new node; < will be ( ( unit + ( norr - tot)) slide cout <<" enter the data for the node"; cin>>temp ->abita; poer -> next = temp; trad (500  $temp \rightarrow next = let;$ S1000+ + IONIN þ At I = NULLY & & (11st -- schota 500 "f (p=='l') list FRCV 500 500 1 while (list -> next! = NULL) Ł 500 17 Idricke  $list = list \longrightarrow reat;$ 1000 list P TOV otra Steer 500 Per . temp = new node; tomo 1000 800 / cout ac "enter the data for the node", mant alle top for cin>>temp ->chita; 51800 10 1500 list  $\rightarrow$  next = temp; 1000 OTTY temp ->next = NULL; 1300 120 AUL 3 ß In lief Void Slink :: delrode () POPU 10001 1000/ q Head Int n, p=0; 1000t BOV node # list, \* poer, \* temp; hup acho > 0 1000 /20000 300 120 1/20 list = head; 1000 could call the data of the node to be deleted: cin>>n; if ( liet == NOLL ) if (head ->data==n) ſ t<<" list is contuin

(a pland + +++) 21+ Ca head = head  $\rightarrow$  next; P=1; } else 1 10 = 20 2000 while ((list -> next! = NULL) & & (list -> atal = n) ſ the still and odds and list the Poer the Poer = hot ; 10001  $list = list \rightarrow next;$ thad 10001 z 10/2000- 120/3000->130/N/1 if ((list  $\rightarrow next! = NULL) \ b \ d (list <math>\rightarrow chta = = n)$ ) 20 = = 20 temp = list; lemp list 2000 2000/ poev=>next = first >next; unit 11000/ P=1; Head 3000 1200 / else if (list->abta ==n) q  $Poer \rightarrow next = list \rightarrow next;$ , old-P=1; delete (tamp); Alla : trank- areat (P = = 0)ş called is not present : ", } z Void Slink: : display() J head node \* list; list ibnar -1000 1000 list=head; ach 25 £12) 10/2000 if (list == NULL) 20/2000 >/30/NUL 0001 2000 3000 Could dist is empty:";

```
z
                                                                                                                                                                                                                                               :+ ~(0)
else
                                                                                                                                                                                                                     il potpibie
           £
      while (list != NULL)

1000! = NULL

2000! = NULL

2000! = NULL

NULL
                                                                                                                                                                                                                                              doact,
                                                                                                                                                                                                                  list
                                                                                                                                                                   Head
              cout << list >data << " <
                                                                                                                                                                  1000
                                                                                                                                                                                                               1288
                                                                                                                                                                                                                                           · APPRO
                                                                                                                                                                                                                 ALLE
               list = list \rightarrow next;
                                                                                                                                                                                                           20/2000
                                                                                                                                                              5/192000
                                                                                                                                                                                                                                           30 NULL
                                                 1000 -> 2000
               4
                                                                                                                                                                          1000
                                                                                                                                                                                                                2000
                                                                                                                                                                                                                                                  2000
                                                  COULSS"NULL";
                                                                                                                                                                                                           utile (option 1:
                                                                                                                                                     2p
            7 Kelse over
                                                                                                                                                      10 2 Mar > 30 2 > 10 Mar > 10 
        5 Matisplay Net
        void maine)
                               -> class dg
          ſ
           Slink s;
            int option;
              closers;
             (not <<" 1. create () "<<endl;
             cartes"2. insert ()"<< endl;
            cout as "3 del node ()' c cendl;
           coute="+ diaplay()"<< endl;
             cout << "5. exit" << endl;
             do
                £
                   coul << "in Enter
                                                                                                      your choice:";
                    Cin>> option;
                    Scotch (option)
                      q
    Case 1:
                S. create();
               break;
      Case 2:
                 S. insert();
                  book;
       Case s:
              S. delnode ( );
                 break;
```

(Ox 5)

Case 4: S. display 1; (10/1 = 1 KEL) STRA boeak; Care 5: it is about - toil as how . break; trate bil - bil } } " JUN " SS JOD while (option !=5); getches; 11:5 ( )alon his 3!=5 41:5 51=5 -> False Slok s: ( ) Receip Colter " s. exit " es endi; course totas guin chairs (nothing << ni) (notion (option) :1 2(0) : ( )960307.2 : Acased

Double linked listi-In the doubly linked list each node contains 3 fields they < d Sinon tabout the ove 1. left link field # and < parcess h> 2. data field 3. Right link field class char , left lips field is used to store the of previous node address 2. Data field; The data field is used to store the data (or) on element 3. Right line field' the sight line field is used to store the address of next node. : Silduri \* In the doubly linked list the left link field of the NULL first node is made and also the right link field of the · Juch = hood last node is NULL. Node structure of the duty litted with left link data sight link field bN field feld contraction prov. Eg: head 1000-NULL 10 2000 1000 20 3000 2000 30 NULL HOULD DO 3000 head = new nede; cates" enter the data for the varie " intobe booker and and the barrent

Program to implement doubly linked list: -- badding #include <iestream.h> boo bit had plated all all # include < conio h> # include < process. h > a kitte that the line line line class Dlink Struct node of blait and that art 1921 day dur contras of performance nate #llink; has a blad alab at blad int obta; node \*strok; ? \* han 1 mode type } \* head; (or) }; node thead; which have be analysis all public: alt Dlink (1) and the left board plant att of int rate is made and also the site list field a the head = NULL; . 1114 3 data 20 z the best where all is another about void create(); void insert (); **VOID** delnade(); Void display(); board Sed 7: void Dlink :: ( reate ( ) Head e [1000f head = new nede; NULL 10 NULL coutes" total the data for the node:"; (in >>head ->data; head  $\rightarrow$  llink = NULL; head  $\rightarrow$  slink = AULL; þ

void Dlink: : insert () list prev  $\hat{\Box}$ (1000) ٤ -temp - annih 500 int n; choo p; note \*liet, \* poer \* Lemp; Muy 5/1000/2 \$500/10 NULL Sco list = head; 1000 whole type rainles variables cast<<" Insert the node at front/middle/last? f/m/1:"; cin>>p; if (p==',e') 9 temp = new node; cout << "Enter the data for the node:"; Cin >> temp -> data; M [500] (500] temp >>olink = head; head  $\rightarrow llink = temp;$ Head Seg terp -> llink = NULL; head = temp; P if (p=='m') Ł verd Diment delade() cout << "Enter the data of a node, before to insert new node. as middle: "; cin>>n; 10 part tistellak gon Poev while (list →data!=n) 1000 500/ 10!=10 temps ? the clater of a mode to be deleted." prev = list; Head 1800 1500+ lest = list -> olink, Inite /5 7 500 temp = new node; 500 6 1000 cout <<" Enter the data for the node: Cin>>+emp->data; per; delete (temp):

Scanned by CamScanner

17 Storen Privales  $prev \rightarrow slink = temp;$ temp -> Ilink = prev;  $temp \rightarrow slink = list;$ ast Poer list -> llink = temp; 500 300 z -Hmp if (P = = 'k')L 1500 5001 -Head while (list -> alink = NULL) 500 NULLI = NULL 9 G list = list -> slink; NULL 5 800 1300/10/15 500 7 temp = new note; cout <<" Enter the data for the  $cin >> temp \rightarrow obta;$ node: list  $\rightarrow$  slink = temp;  $temp \rightarrow llinks = list;$ temp -> slink = NULL; void Dlink: delnode () q >> 200) int o, p=o; to the note \*/11st, \* proev, \*temp; list = head; (a : Lotator - Lett) 91262 cout << " Enter the data of a node to be deleted : "; cin >> n; if ( head ->obta==n) emp 15001 500 S 5 = = 6 500 temp = head; NULL 3 800 500 6 1000-150010 NULL head = head -> slink; 500 800 head -> Mink = NULLA 1000 Cho>>>temp-fdate; P=1; delete (temp);

} Sool = NULL : "Epignos 21 tall" >> tool else 1 while ((list  $\rightarrow dink != nUU)$ ) & (list  $\rightarrow data!= nUU)$ 1000 = NULL Ł middle prev=list; cast ac' MULL "ac " as "; list = list -> vlink; (unh - 1 tot) - dill z 1000 / = NOLL if ((list -> rolink!=NULL) & & (list -> data == n)) ð 1151/0temp = list; 500  $prev \rightarrow slink = list \rightarrow slink$ ,  $list \rightarrow slink \rightarrow llink = previ$ tempin P=1; Head 500+1 delete (temp); ( jaion biov f NUL 5 200 clee if (list  $\rightarrow data = = n$ ) 156300 ſ n temp =list; 500 poer > reline = list > rline := > () have the (col) rast car (191). delad ()" << endl; P=1; plete (temp); :1603 >> 100 à 1921 b. + ">> 100 2 (p = = 0)AUNS 800 q cout << "Node is not present : " CIN >> option; j 2 Switch (aption)) Dlenk :: displaye ) Void q 11 2800 \*list; node d. Conster); lest = head; : Areard 12 (180+ == NI) 1) 7 CASE

S Cout << "list is empty"; ordina. } 100 (units - ) alich ( ( the - ) alich - ) the ( the - ) the ( the - ) the ( ) ) else 500 £ caut < " NULL" << ", 500 1000 1000 1000 1000 Lotile (list ! = NULL) z ((is = atob < til) f.f. (un = ! anila < ter)) i cout <<" list -> data << " (=> ";  $list = list \rightarrow slink;$ NUL 4546 6 1000 1500 · dails - teil = dails -Cout << "NULL"; Wand = HOIN - HOIDE - HI 7 void main() delete (temp)) ſ dink d; int option; Closcoc ); Cout << " 1. create()" << endl; tem - list cout << "2. Ensert ()"<< end); "< end); "< end); cout <<"3. delnade ()" << endl; delete (temp); cart <<" 4 · display( ) << end. 1; cout < c" 5-exit" < codl; dof cat << "in totor your choice:"; a short ">> koo CIN >> option; Switch (option) ł Void Dink :: displaye Case 1: d. creater ); \* list; shoa break; list - head; Case 2: (LICH == NOLL) d. inseat();

break; case 3: the circular both and and the book into the two d.delnodec); break; tell botal signe kinde linhed list; shoed d. display(); harall Block introductions case +: NULL Entry Light Son this ends not shall be the ends man wo fields they one. I. Rota field case 5: 4 tota link (a) while (option! = 5); field next field getch(1; the fight the Data field can be used to store the to (or) on element to get it set to the line field can be used to the address of next node. s the circular single linked list the last node link relates the address of first rade in order to the linked list as cliecton Head post Sib 1000 1000 10 1000 0002 01 0001 20 3000 30 1000 0031 2000 3000

the state little for the thirds the

( Armel Circular linked List;-The circular linked list has been classified into no types they are 1. Circular single linked list along rase + : 2. Circular doubly linked list. 1. Circular single linked litt' In this each node contains two fields they are. 1. Data field 2. link (08) next field. Alate Staucture 1-:(e: 1 millon) 31idu Data link los) field Dest field Stary: Data field's The Data field can be used to store the obta (or) an element link on the next feld - The ling field can be used to Store the address of next node. \* In the circular single linked last the last node link field stores the address of first node in order to make the linked list as circular Eq. TO Head Egio Head 1000 1000 10 1000-(b 2000) 20 3000 30 1000 1000 2000 3000 2. Circular doubly linked lift - In this each node contains 3 fields they are left link field, Data field, right link field. Note structure:link Data Sight link field link field Left link field! - left tink field holds the address of poevious node.

pata field; - Data field can be used to store the data (on) an element. board line field to Right link field holds the address of Right eisters the data for the next nate con >> ho \* D the circulon doubly linked list the last not link field Sight Stores the address of first made and the that note left link field stores the address of last node, In order to make the linked list has circular. Eg'r O EJO head 1000 ig norb head 1000 tail-\* short 1000 10 1000 3000 10 2000 1000 20 3000 2000 30 1000 1000 1000 program to illustrate circular single linked list:-In linked list it forms noits the nodes of #include <iostream.h> addresses are processing that is the # include < conto.h > reason we are using epocess his # include < process. h> temp : new mode; class clink 4 struct node: : down adde of notobe add making"> cin >> temp - + data: temp -> nont - hend; ift data; node \*\* next; i grant + tran + - brand ] \* head; gant bist public: colink() ('m' == q) 1) head = NULL; inters the data for the order believe for insent Void create (); new male as void cin >>n; insert (); delnode (); Void (a=lotob + sell) strike display (); Noid 1;

3 [100 to boot of all void cslink::create() -transla Ł 2 10 1000 noo head = new node; black april 1000 cout << "Enter the data for the node: "in the tool and the back plant the restants and the head -relation = head; million and enote high mill the noch the first rate left into feld plans Void Colink :: insert () list Prev, botemptor 1000 PT Con CO State of the little has grateled [900] int n; char p; read node \* list, \* prev, \* temp; list = 2000; 200 000 29/100 200 000 coutes" Insest the node at fort/middle/last ! f/m/1: cin>>p; # include < instarcom h> it (P = = 't') + indude < imito b = # include < parcess. ha temp = new Node; class clink could' Enter the data for the node: "; and burge cin >>temp ->data; temp -> next = head; int data; that we was next. head  $\rightarrow$  next = temp; ibood\* 1 head = temp; Diduq () HNIED if (p = z'm')ş HULL: bood cout << " Enter the data -los the node, before to insert new node as middle in biov C'in >> n; 10 void insert (); 10 = 10 void delade ();  $\text{while}(\text{list} \rightarrow \text{data}!=n)$ il o polopito biov ٤ :1

```
poev=list;
    list = list \rightarrow next;
                                  head
    2
                                 black - bland - broad
   temp = new node;
    cout << "Enter the data for the node:"
    in >> temp \rightarrow data;
                                           delete (temp);
    prev \rightarrow next = temp;
    temp \rightarrow next = list;
     7
    if(P = = 'l')
             iddie (list + next! = hend) & & (list - + dda!
    .L
    while (list ->next! = head)
                                        +==1=V360
     ş
                                     that fill :
     list = list \rightarrow neat;
    temp = new node; be toil & & (bord - i tone toil) "
    cout <<" inter the data for the node:
    Cin >> temp ->data;
    list -> next = temp; when to it = hand - Varg
    temp -> next = head;
                                         delete (terp);
    þ
  7
Void cslink: : delnode()
                                  (is if (list - ration = n)
   5
    int n, p=0,
                                              temp = list;
   node #list, #prev, #tomp;
                           tent tell: trank - barg
   list = head;
    cout<<"Enter the data of a node to be deletered : ";
                                         delete (temp);
    Cin>>n;
    if (head \rightarrow data = = n)
                                               (0==0)
       temp=head;
      while (list -next != head)
```

f (Bet - test - test list flist -> next (abor win head=head ->next; list -> next: head; and alob add makes ">> how 12rd The p into + - gaint < < nit P=1; istual - teast - hard delete (temp); temp -> next = list, 3 else utile ((list  $\rightarrow$  next! = head) & & (list  $\rightarrow$  data! = n)) (hood = ) theory - toul) olidu £ prev=list; that = list -> real : list = list -> reat; 7 if ((list -> next != head) && (list -> data==n)) call ex" inter the data for the na temp =list; Cin >> temp. ->dato: poer->next=list ->next; 191021- 10001-P=1; m'ddle rode temp -> reat : head : delete (temp); clse if (list -> data == n) Void celinist: deloude() Y temp = list; est nobe ignoix (varger, tille aboo poer -> next = list -> next; Cout <<" Entry the ctata of a made to be; "Effectended" ibnata tell delete (temp); (desoi) (n== state (-board) # 1/else "( p==0) 2 while (list - rest intend)

```
cout <<" nlode is not present: ";
  }
                                                         :1 36000
 7
                                                 (Souther);
   Void Colink :: display()
                                                      ; HORE
   ł
                                                          case 2:
   mode *lac;
                                                :( )tracai.es
    list =head;
                                                      ; HORA
    if(list = = NULL)
                                                          S 200
      Į
                                               (c) delnode();
     cat <<" list is empty:";
                                                     break;
      Z
                                                          cale 4:
   else
                                               :( )Koldsip .S)
     q
                                                        : XPERK;
      while(list \rightarrow next: read)
                                                          : COSC 5:
        ſ
                                                     ; ADDad
       cout << list -> duta << " ( );
        list = list -> next;
     fout << 12st -> atta;
                                           While (option 1:5) )
Void main()
 5
  cslink
         cs;
  int option;
  closers();
   cout << "1. create ()" condl;
   (out << "2. insert ()" << endl;
   Cout << "3. delnode ()" << endl;
   cout <<" 4. displaye) 'exendl;
  Cout << "5. exit "<cendl;
do
 2
  cost << " in Enter your choice: ";
  cin >> option;
   switch (option)
```

i have not is not present i Case 1: (S. coentel); Cherdon :: Horkey Hork break; case 2: tilk short () traedices imad = tell break; (LICH = - NOLL) ?! case 2: Cat <<" Here is compty:"; cs. delnode(); break; case 4: (S. displaye ); boeak; (hand . ! tran - teil) slinks case 5: boeak; "Los'ss of the tril so tool the - life - next : 7 fort collist -> dilai while (option ! 55); getch(); ( min blow cslips imityo tot 11 102000 cout es "restate ( )'econdl: (lbrows '() trach () ' econdly (out <<" s. delinate ()" <<end); Cout <<" +. display () ' << endl; Cast 25 East "20001; :312

Search ) operation for the single listed list; -flag void Slink: : search () pos ita listiav q D 1000 int n, pos=0, flag=0; (0= 209 NOLLINI not \* list; Head list = head; Moof list : head if (list == NULL) \$10 20001 20 3000 +>30 NULL 1000 = = NULL 1000(110 Ş 3000 cout<<" list is empty"; cout <<" list is empty: 3 else 2219 couter the data of a node to be searched, Cin>>n; 1000 = NILL (not extenten the data of a rade of while (list] = NULL) While (rist ! = NULL) 9 POS++; ++++209 if (list  $\rightarrow data = = n$ ) 10 = = 30 (n== abobt- tail) \$ q 36 = cout << /ist->data << "is found at: " << pos << endl; (art edits state of "is found at :" adopt technolis (1= golt list = list -> nedt; de: 10 is formal at : 1 [ 30 is -lound at : 2 [ 20 is -lound at : 3 [ (0=:2pol b) ] Mushile ju //else if (flag == 0) cout << Noble is not present with the given eliment: "; Could as not present with the given element it. J (Iscarch()

Search, operation for dubly linked istipostor plag anily b n void Dlink !! Seatch() 20 0 0 1000 ş : 0: polt, 0=80 int n, pos=0, flag=0 Head node relist; 1000 list = head; - Inutiol2cat. Flood 20200 Jasso Nal 1000 = = NULL if (list = = NULL) 3000 ę cout <<" list is empty; cout << "list is empty", P else could find the data of a nade to be seconded; cout < "enter the data of a node to be scatched: Cin >>n; 20 while ( fist ] = NULL) while (list!=NULL) 1000 ! = NTULL ą 2000 - NULL post+;  $(0 = -nth \ll$ if (list ->data ==n) cat as list -> data as "is "fand of :" as pro condition cout < d'ist > data < « "is found at : " << pos << endl; flog=1; : 北のく やけ = やり list = list -> vlink; if (flag==0) (0==polt) li interfects not present with the given eligent: " couters "node is not present with the given element:"; dpit 20 is found at: 2

Becorch() -br circular operation linked listi-Void Cslink: Securche ) Ed Tol (Et flog Loop intern A+ K 4 2000  $hinton, P=0, \pm log=0;$  hill pride in the prior of theantan node in + list; about aliptiopoot 730/1000/ list = head; h doro] (1)0 3000 11 if (list = = NULL) \* using the hashing technique the 2200 \* mitrogo 000 2 manual is empty:", with par barrow ad dig phonon elsemply will a pinton and to anon and filmancel seenth etc. Ł cout << "Enter the obta of a node to be searched:"; cinson? and an an upindut proton in 07 while (list -> heat! = head) himsen o homsen dictionally - A dictionary is ed of relar (00) 8 POS++; aipto)  $if (list \rightarrow data == n)$ isto "thinkno arpintoa) mintan art \* 197 cout < list -> data < <" is -found at node position: "<< kno naitald . to part to back ald a possered; 23 flag=1; 7 ability dept 1 list = list -> next; apitant doubt e f 11 while S. (ollision °f (list →data ==n') 2000 -- 20 = - 20 POS ++; coutedist -> data <<" is found at node position: "<< pos O flog=1; when the set of index index is ot 90012 most//letse ate transla no sinter (an at node poetion: e xH. hoch lotte. if (flag = = 0)at \* not present with the other element: ". 1-h of cout << " Node is present with the given with 10 420 970 23 slided dear att.

Hashing i-\* A hashing is a technique which is used to perform certain operation like insertion, deletion and search opesation on a specific data structure with an average constant time o(1) [order of 1]. \* using the hashing technique the search operation can +41 ">> be performed very efficiently when compare with some of the technique like linear, binary search and fibonacci seasich etc. cast <<" Enter the child of a node \* The hashing technique can be used usually to implement a special data structure is known ay dictionary. A dictionary is set of elements (or) contain. (n== ntob <- teil) \$1 \* the hashing technique contains are uses three key components in order to pertion insertion, deletion on search on a costa staucture. They are. 1. Hash table list = list -> nert; 2 thack -function mapping 3. collision (in == ntob <= tail) ? tosh toble:-: 1+ + 309 \* A hash table is a clota structure which contains a fixed length array with set of index values to store (a) search (as) retrie an element sto (as) from the hoch table. (0==pol?) fi the index of the hash table stools from a to N-1 where N' is the size of the hash table.

Host -lable (6) Aprilsion Hethadin D o to N-1 \* In this method, each key clearent w be mable alwing 0 by had, table size and the tanking volue coo to 4 (10) reached as hach address (10) 1 10 ceasion 5 nI gratu sklat deal \* In the hads table each key element mapped (or) à restantion associated with the corresponding index value of the hash table in order to perform the operation insertion deletion and search on the hash table. Hash functioni-. oste abot deord = M \* the host function is a function which can be used to map the key elements with the corresponding index values of the hash-table to perform the operation (insertion, deletion and Search) \* The value which is generated by the hash function can be called hash cade (00) hash address (00) index of M the hash toble. \* In this method eath they based L'Division method at la suie all no que pribrages 2. Mrd-aquare method bivib alubar ad 1100 did alubar 3. fading method and be considered and the bar Multiplication method and to contable million (m) ". The hash function can be denoted by 'H'. 85,3121,3 = 8 mar

Division Method ;-

\* In this method, each key element will be modelo divide hash toble size and the remainder value can be Бу treated as hash address (as) location address of the hash table where we can insert ( or ) delete ( or ) seally the posticular key element into (a) trom both table \* The general form of this method is about door to realism the reitizini northize) - H(k) = k mod m detetion and search on the hash table. Ŧ where K = Kay ¥ M = hash table size. 25 Hash toble sit 1 20 fa'as males antenat a si asil xst. 20 623 Keys = 20, 28, 32, 44 120 rebai 22 <u>32</u> values 20 H(20) = 20%10 of schiddle all 44 H(28) = 287.10 = 8 box m5 6 atr 4 H(32) = 32%. 10 = 2 mg  $\frac{1}{100} \frac{1}{100} \frac{1}{100} = \frac{1}{100} \frac{1}{100} \frac{1}{100} \frac{1}{100} = \frac{1}{100} \frac{$ be colle 28 E the hash table. Mid-grace Methodi-\* In this method each key element his squared and middle bits are extracted from the squared number depending up on the size of the host table and also the middle bits will be madulo divided by hash table size 6 and the semainder value can be considered as hash address (or) location address of the hash table. \* 4 n-the hash function can be denoted by bi=m 6 F Keys = 6, 72, 15, 32 C

 $H(K) = K^{2} = 6^{2}$ 

ملحط دهدا لحد أور = 36%10 = 6 D (12)<sup>2</sup>= 5184 40 = 18/·10 = 8 1001 = m = 100 15 2 3 N=12456 (15)<sup>2</sup> = 295 / 10 = 97/10 = 9 38 12 34 56 (38)2 = 1444 ×40 = 44%10 36 21+34+65 72 Them Folding Method: This method is divided into two types They are infold chifting method wight added hallow 2. fold boundary method briten and at Fold shifting methodi-\*In this method the key element is divided into equal postitions based on the hash table size and all the matrix the postitions will be added and the sum will be considered as the location address of the hash table if any carry is generated while performing addition operation between all the postitions, the carry will be ignored. KA = 20×0.618033 Si m=10 = 12.36066 K=123456 = 1+2+3+++5+6 2000 0 x 01 = [AZ] = CON location address of hash table county is ignored hand elevent bandary Methodistion of the and hearing at fold the proceedure of this method is exactly similar, to the stold potsilling method, exp except that the fight and 1-21 : A gillor hotos bet postitions will severed and also all the possibilitions will be added and then the sum will be considered as hash address it any corry is generated

that will be ignored. 0 : 01×08 . 8 = 01.481,0 01x + 812 - 4(47) spi - m = 100K = 12.3456 88 P OR = 12.44 E(21)= 12 34 56 18 01×++ = 01× = 0.618033 - (88) -= 21+34+65 = (120 ) - C > sum equit occurry des lighting. et bottom 20 x 0. 6180331 priblet Multiplication Methodiation publido: 0136066 3 Praction to In this Nethod the key value is multiplied by a constant value and from the result the fraction post will be extracted and that will be multiplied by siz of the hash table and from the final result the integer pad will be considered as the hash address addition m=10, K=20il any coursy is generated while performing addition openation between all the proditions, they canad gd Illa . berapi A = 0.618033 or-m -ip KA = 20×0.618033 K=123456 = 12.36066  $m[KA] = 10 \times \mathbb{Q} \cdot 36066$ alt 6066 lo control 109 -Parate si hread The general form of this method is [H(K) = m[KA]] This method will work efficiently by talking little possible will be added and then the sum will be 1 and

collision !printing to implement chairing the collision is a problem (or) situation which will be occurred when more than one element need to be stored into the same location address of the hash table. collision problem can be avoided by using two types The at techniques these are 1. open hashing about bust 2. closed hashing hashing :- This technique can also be called as open chaining (or) seperate chaining this technique can be used using linked lists to readire the collision problem by bid initializebostable() m=10 Egit K=10,20,28,38 toi : Unincitation f collision is accured biov 0 · ( ) tosens 10 I biov H(10) = 10% 10 = 0 realizion H (20) = 20/20 = 0 ( ) prologit 2 biov 3 H(28) = 28% = 8 ] collisionH(38) = 38% = 8 ] collisionbov 4 5 6 J collision percented to biblion i condo biov 28 8 hi a calision m=10, keys = 10,20,28,38 20 NUL (++i (2000);0:i) wit 10/2000 0 1000 1600 2000 H (20) = 20%10 = 0 NULL (100 H (28) = 28% (0=8 2 NULL +(38)=38% 10=8 3 NULL NULL 4 (a tri) autoreal : night thi 5 NUL 6 NUL Seturn (R. Smar); NUL 38 NULL 128 2001 8 1001 1001 2001 NULL (b) collision avoidance () trach :: initial bion

Program to implement chaining # include " liost ream. h> makter of a ci moi illo atr the state conio. h > and and soon and have been 1000 C # define max to constant without and alt all the collision particles can be avoided right while the collision have tout node prideor and i ago i soot to engineers prideor bacolo . c d allo nos supiralistations technique con alla int data; beeu al ] \* hashtable [max]; pinints storages (00) prinials Public: prize und mathering militility and analy of void initializebastable() several of or int hashfurrint; to f collèton is acroad void insert(); Void delnode(); 0 = 01 Noi = (01) H Void display(); 0 = 01 Noi = (00) H Void Seatch(2); 8 = 01.(88 = (86))þ; void chain: initializerashtable greater as Ł (a) (ditsion it i; m=10, Keys=10,20,28,88 for (1=0;1<max; i++) 0=01.V01 = (a) 1 0001 hashtable [i]= NULL; JULA JUM 8:01:85 LIGH H (38) = 38×10 - (8 JULA int chain: hashfun (int k) MOLL E 1 KAA 10×10 20 20%10 00 1 KAA betum (K iman); 1001 7 JJCH1 (b) cellision avoidance Void chain:: insert() d

int pos; node \*temp , \*lict; -tombros pos terrist temp new node; : trank cout <<" Enter the data for the pode in the cin >> temp -> data; -1000-SolNat Juli NUM temp->reat= NULL; - (HOCO 9+9280 POS = hashfun (-temp -> data); 9819 if (hastable [pos] == NULL) ş while (temp - Sneat! = NULL) & (time ( hashtable [pos] = lemp; 7 i empt = terrp ; else itime quet : quet : ٢ list=basttable [pos]; While (list -> next != NULL) = i trace quot) } theart = trenge -> meart; list = list  $\rightarrow$  next; þ if (let ->next == NULL) delete (temp); 5 list -> next = temp; (a== ntrbe-good) \$ -2013 2 12:1 Void chain: : delnode () ; hone-grapt - tron d delete (tom); int pos, n, p=0;node \*temp/\*list; cout <<" there element to be deleted: "; tomp [ list nr un>>n;" POS 10 01 1. 1. WON 1000 0 2000 10% 10 :0 pos = hashfun (n); band percent (27/200) Jon sil 2017000 3000 3000 temp=hashtable [pos];  $if(term \rightarrow data = = n)$ 

d the rock; list = temp; i toilet, ganatit above temp = temp ->next; temp up - not hashtable [pos]"= temp; also alt alt alt alto P=1; ( atom - anot - < n's delete(list); Whith there and 1 i (otabe - ganot) mikkoni = 200 else if (hashtable [pas] == NULL) f uchile ((temp > next!=NULL) && (temp > data!=n)) list = temp; temp = temp ->next; 7 [em] adated= til  $if((temp \rightarrow reat! = NULL) \notin \& (temp \rightarrow data = = n))$ list  $\rightarrow next = terop \rightarrow next;$ P=1; (Isola == heart == NOLL) delete (temp); ignort + hear a temp; else if (temp=>data ==n) 5 list ~ next = temp ~ next; () should !: much bion delete (temp); int posin, proj 3 7 hode #tempty-list; it a (plate of the element to have of the ">> too :0<<01) couter" n 22" is not found at position; "<< possended յ Դ temp - hashbalde [pas] ; if (temp -> dota == n)

void chain : display() to bout to a set so too £ int i; node \*list; ( John Hidr for (1=0; icmax; i++) £ riph list = hashtable [i]; copitad (if (list == NULL) e Couter " list is empty:", identified it "> cartes"2. hoese ( 5 X (lbno>> while (list ! = NULL) (out < <" 3. dlade () "<< 8mdl; ł Keys = cout << 113+ -> tota < c" 16", ?>> " 0 1000 - 101 WILL 11 } ۲ : lbdg >> 5 Cout << "ventery chain:: Search() void roof int pos, k; node -relist; cout < c " enter the element to be inserviced: "; CIOD>K; 960) Pas = hashfun (K); : () skintertys il pitini .) i aparol list=hashtable [pos]; Sale) while (clist != NULL)  $f k(list \rightarrow data != K))$ () toppai .) q list = list -> next; SIGOK ; Calk 3: 0 1001 - FNULL 1 = 1 if ((list!=NULL) & & (list -> data == E)) ł DREAK; coutcelist -> obta ce" is found at position Pos is player; j. çlæ <cendl; break;

costecher "is not found at position" << possendly 3 3 ini void maine, : tilk tox 1 (++i: pornsi (0:1) to chain c; int option; ([i] aldothend = toll closer ); (IJOH = NOLL) cates"1. Initializehashtable()"exempl; 10">>>100 coutes"2. insert ()"<<endl; Cout << "3. delnode () "<< sendl; (ijoh = [ teil) alkter Couter" to dieplay ()" excende to obtain hell so has cout <<"s. search()' < cendly theme with this COUT << "G. Exit" << endl; do £ Cout << "venter your choice:"; cin >> option; ( N , 209 Switch (option) Ł could are the element to be freenoched: Case 1: Cino >K: c. initializedastable (); (2) where 209 break; : [209] eldorideord -teil Cale 2: ((x · ! n lab < - teil) \$ \$ (LIVIA = ! teil) ) slicks c. insert (); break; : trank teil : teil Cafe 3: d C. delmode ( ); b = teil) & & ( LUM - ( teil) ) 7; break; Course (ist -> dota es" is found +: c. display(); si bna>> break;

case 5: c. Searche 1; break; cose 6: break; 3 f while (option !=6); geten ); f