```
linked list
  # include < stdio. h>
  It include 2 stable, h7
   Stautrodec
  interior into
  Stewed node * Jink
  type def struct node* NODE;
  NODE getrade () {
  NODE X:
 X = (NODE) malloc (Size of (struct node));
  if CN=NULLD &
 point ( comemory full \n");
 ent col;
  outron x:
void free node CNODEX) {
 Jee (x):
NODE insert front (NODE first int item >8
NODE temp;
temp = getnode ();
temp > info = atem
temp - link = NULL;
if (freit == NULL)
galven temp;
temp > link = just ;
just - lamp.
areturn just:
Node delete- front (NODE just ) ?
NOOE lemp:
```

```
if (first == Null) {
special ("list is empty connet delete ").
Setura first:
temp = fiest:
 temp = temp > link;
 pointy or item deleted at great and is % of 100
    ic spire - ties
Jeco ( first );
 relivin temp;
I ( note treat about ) rener tream about
 Node temp, wer'
demb = get node ():
 temp > info = item;
 temp > link = Null:
 if Cjinst == Null)
 sietuen temp;
 cur- juist;
 while (curo- link 1, = Null)
 Cur = eva > link;
 Cus -> Jink = temp;
 return just;
 Node delete - gear CNODE first JE
  NODE CUR, PROVE
 36 llus == Lauf) fi
 : ("n' etelde tannes yetyme in tail") thing
  Elief neutore
  if ( first > lik == Nell) 5.
 perint (" item deleted is % d\n" first > enpo)
```

free (just): relian rule; Perer = Null; cur = josti ushile Can > link! = NeulD. { Ples = cusi Cur - cur - link 'special Citem deleted at rear end is % d" (w) Jace (cur) pero > link - rull: relien fast; NODE enset Pos cintilem, ent pos, NODE field NODE temp, cur, Press; int count: lemp = getrode c); lemp - singo = stem; temp >> link = NULL; if (first == NULL& & Pasz=1) { relien lemb: if (fust == NULL) & paint (" invalid position \n") relian just; 3 (Pes == 17 & demp -> link= jerst: fieret = lemp; return temp;

```
Count = 1
- boor - Null:
cur = first
copile ( cas! - Nell & & count! = Pos) {
per = cur;
Cur = Cur -> link
Count ++;
if C count = = Pos) {
prece > link = temp;
temp > link = auri
Outeun fint:
parinty (" invadid position \n")
return just;
Node delte- Pos (int Pos, Node fint) 5
Nodo curi
Node Press;

int country flag = 0;

if Cfrit = = Null 11 Pos < 0) [

pountf ("Invalid position In");
 relien Null;
 HCPOS==1) E
  Cuer = first ;
just = just > link;
 Jerensode (us):
 itary neutone
 pour = Null
 cur = con link juist
```

```
Count = 11
  ciohile cours = Null ) {
 if Count = Past E
 flagels
 decaki
 Cout ++:
  Page = are:
 are = are > link!
 4 Glag = = 0) 1
 so tainty ("Invalid Postion ("))
 Pourly (" item deleted at a given Position is yed)
     are Junto);
 pace > link = are > link :
 force node ( cua);
 exturn foist:
  earl display ( node first ) &
 None temp;
if (fist == NUL)
Point of color empty connot display theme in for (temp = first temp! = Null interps
 penty con of in temp > info )
roud main ()
int item, charce, Key, Pos:
int count = 0.
```

```
NOPE Just = NULLS
1 (··) 80/
friend ("In!; Smeet seren 1,2: Delate hour In3:
expertisent 194: delate front 1 05: most repo
Bulyalquib: Fal naited afri stelds: 301 nailed
 · Exit (n"):
: (" Enter the chaire: ");
Scanf [" % d" & choice );
Swatch (choice ) {
Case 1: Paint (" Enter The item at seas and In").
scan ( " % & & Litem );
 Emple , treef ) ease _ time = tiest
 beak;
 Lase 2: Jost - delete = sear ( Just );
case 3: print (" hEnter the iten at front end (n");
 scanfly or d, liters);
just = west - front (first stem );
 break'
  cose 4: Just = delete = Jeant Cfrist 2;
 beeak:
 case 5: printy (a Enter the item to be inscribed at
 a guien Pointien (n"):
 Scan ( " % 1 x, b item);
 print (" Enter the Postion In")
 Scanf (" o/od", LPOD).
  fuit = mout : Pos (item, pos,
  beenk;
 Case 6: Print (" Enter the Postion \n").
 Scan ( " % d" & Pas);
 Jast = debte Poscipos, just )
 · sperch
```

cope 7: display (just);

Jessel;

default: lind(0);

break;