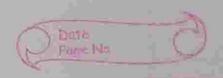
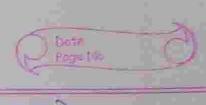
```
Lab-5
             Linked list.
Hadube & stolio. hs
# include < stollab. b>
tract node
int or fo;
 struct node * link;
typedet struct node * NODE;
 NODE getroode()
  NODE X 0
  X z (NODE) malloc (seze of (Struct rude)).
 if (x= NOTT);
 printf("men full \n");
 6 CXE+ (0);
 return x:
void free mode (NODEX).
free (1)
```



NODE Engent front ( NODE frenct, out stem) NODE temp; temp 2 get noder); temp -> Solo z item; temp - s link = NULL; of (forst = = NOLF) of return temp; temp = forst: temp = temp -> dark; pointf (" Hen deleted at front - end is = 1.010", flogt -> Enfo); free (forst); return tempi NODE INBERT REDUT (NODE FERST, Int item) NODE temp, cur; temp = getnodel): temp -> info = I tem; temp -> link = NULL; of (first = = NOLL) return temp;

Clur z forst! whele (cur ) link 1 = NOLL) L cur = cur-> Link; 4 cur \_ s look = temp; return forst; NODE delete-rear (NODE first) NODE com, Doen; Ef ( forst = = NULL) prints (" list is empty (n"); retard forst: of (forst -> lank = = NON) prent f (" Etem dokted us /d | n" forst-sigo free (forst); return nutt; poer 2 NULL; cur 2 kirst;



while (cur -> look & = NOLL) Proev = Cler; cur = cur -> Linda o posite ("Item deleted out rear end 1-dis) , cler - into); free (cur); prev - s dende z NULL: return frost: vold Dosplay (10002 forst) NODE temp; Of & (forst = = NOTT) of poontf" (" lest empty io")} for (temp = forest; bergp 1= NOLL) temp = temp-)(sh) printif ("ofod lo", temp-> info);

vosal mach () ant otem, choice; GODE COUST = NOTE: prontf ("10 1: Ensent-front in 2: Detete front in 3: Insert - record 4: Delete - rear (on '50, Dosplay to G: Exet (n") printf (" Poter choice (1"); scont (" of d", & choice); Switch (choice) Cose 1: point + (" Enter Hem 10"); break: first z Engent-front (first item); preced. Case 2: por frost = delete front (first); lo reak; Case 3: pointf ("Enter Herolo"); forst = insert - rear (forst item); break o

