```
22/01/2024
         hinked hist:
#include < stdio. h>
                                    Fronte (chorce) = 13);
# include < malloc. h >
 Struct mode
        int data;
       Struct node soment;
  Struct node astart=NOLL;
 Stand node & create-ll (stand node &);
 Struct node & display (struct nodes);
 Struct node , insert-beg (struct node a);
Struct node & insert _end (struct node 2);
struct made & insert-before (struct modes);
Stant node & insert_after (Stant node a);
Struct node & insert delete-beg (struct node =);
struct mode & delete-end (struct made &);
Strutt node & delete mode (strutt node &);
Struct node & delete-after (Struct make &);
Struct node & detete-list (structurade &);
Stonet node #Sort-list (stonet made #);
Stand node a sort list (stant node a);
int main (7
{ int choice;
     print f (" no 1. create a list", "2. Display", "3. insert beg;
"H. insert_end", "insert-before", "6. insert-after");
    Print f["7. del beg "8. del rend It 9. del mode 10 del after
    It 11. del- list \t 12. Sort_ list \t 13. emit \n");
    Scanf ["-1.d", whoice)
     E cose 1: Start = Greate-Il (start):
                  P& L" linked ilist cereated ");
         Case 3: Start = poinsent - bog (start);
break;
```

```
Corse 4: Start = insert end(start);
} while (choice! = 13):
returnos
Stomet mode & create-U (stomet mode & start)
    stand mode sanew mode, aptri
     int num;
   printf("enter -1 to end");
   prients ("enter the data:");
   Scanf (" yod", &num);
   while (own 1 =-1)
 E new-mo de = (stomet mode à) malloc (size of [stomet note)
new-mo de > data=num;
              if (start == NOW)
             2 new-mode-soment = NULL
               while (ptr-nent! = NULL)
                  ptr=ptr-ment;
ptr-ment=new-mode;
                  new-node -> nent= NULL's
               pointf ("enter the data");
         Scanf ("/d", & num);
           } return start;
```

```
stand mode & display (stand mode
E stand made aptro
   While (ptr/=NULL)
    { prints ("It %.d" ptr-)data);
ptr-ptr-ment;
     redurn Start;
Stonet mode * insert-beg (stonet mode * start)
    Stand mode a new-mode;
    int num;
    prints ("enter data i");
   new-node = (stmit node *) mallo ( laize of (stmit
     node));
    new mode -> data : num;
         node -> ment = start;
     Start = new-node;
     return Start;
Strut node & insert-end (struct node & start)
    stonit mode someno-mode, septro
      printf ("enter data");
       Scarf [n.1.d", knum);
      new-mode = (stanstmit mode &) malloc (size of (
       new-mode -) data = num jug
       new-mode-ment = NULL;
        pt~=Start;
      if (stant = = NULL)
         start = new-mode;
    ches
```

while (ptr-ment 1=NULL ptr=ptr-> nent; ptr-> nent = new-mode; return start; struct node a delete beg (struct node ostaet) stand node spt ? ptr=stant; start=Start -> nent; free (pto); return start ? Storet node odelete-end (storet mode o storet) stomet node aptr, apreptris while fotr-neut != NULL) E preptriptri prept -> ment = NULL; free (pt r);
return start;
}; Stonet mode a delete list (clamost met mode astart) stonet node optri ptr:start; wull) E printf ("/od is deleted", pt >-) data); y; I return start;

Stort node a Sort-dist (stond node a start) E stand mode \*ptor1, optors 5 int temp; ptri-start; while (pto 1 -) nent (= NULL) E ptoz=ptor-ment; while(ptoz)=NULL) {

if (ptrl) data > ptr2 ) data)

{

temp = ptrl ) data;

ptrl > data = ptr2 > data)

ptrl > data = ptr2 > data) ptr2-ptr2-ment;

ptr1=ptr-ment;

display(start); P.E. Ent )
22/1/24