Hedra

## papergrid

Date: / /

```
Lab Program 5
struct rodo i
     int data;
    struct node *next;
roid insertint ( struct node + theadptr) }
   extret node * newnode;
    int value,
    printy ("Fater value: ");
    scant ("1.0" evalue);
    neunosse = (struct node +) mostor (size of truct node),
    newsocle -> date = value;
    neurode -> next - NULL;
    if ( + headptr == MULL)
       (*headpty) = newnode;
     else I
      neumode > next = ( + head ptr);
       ( + wead ptr) = new node;
     z
 3
void insert pos (struct node ++ headptr) 1
     surer node + newnode, + temp;
     int count = o, curipos = 1, value, pos;
     print ("Entervalue: ");
     conf ("1.d", broker),
     newhode = (smeet node +) made (esized (struct node));
     remode - dala = value;
    newrode = next = NULL;
    y ( + Loapt == vac NULL) {
            ( * read ptr ) = numode;
```

- }

```
papergrid
                                        Date: / /
case 2
  enp = (+ headptx);
  while (temp! = NULL) &
   temp = temp > next;
  print ("There are ". I defended in the list. Enter position").
  scanf (" 1, d", & pos);
  if (pos > (count+1)) }
   mint (" No was possesson (");
  e relain'
  if (PO) == count +1) {
     temp = ( * headpr);
      while (tam p -> next ! = NULD)
        timp= temp = next;
     temp - next = newwork;
     temp = ( * headptr);
     while ( semp - next! = NULL) }
       if (curpo) == pos-1) &
         newmode -> next = temp -> next;
           temp > next = new node;
         Curro + -1;
        limp = timp + next;
```

## papergrid

Date: / /

```
void insert look (struct node # headpt +) à
  Struct node + new node, + temp;
  int value;
  printy ( " # Ole value");
  scarf (" 1.d", b value );
  responde = ( truck node *) malloc ( rised ( ranch node));
  newhoole -> data = value;
  reunode - next = NULL,
  temp = (* neadpty);
  if ( the adpris = = NULL)
    (* headptr) = neumode;
  else s
     while ( stemp = next != NOL)
       temp = temp > next;
      timp = next = newhood;
  b
void deser last (struct node it headptr)?
      grance node + exmp;
      semp = ( + headptx);
      if (+ headptr == NULL)
       prints ( "The list is emplyin");
       else if ((+ headpta) - next == NULL)
       ( + headpt ) = NOLL'
       else 1
          temp = * head prr;
           while ( ( bemp > next) > next ! = NULL)
             temp = temp = next;
         tem p-> next = NULL.
```