## LAB PROGRAM-8

WAR to implement Stack and Queue using Linked Representation

NOTE word seems ( note forest, with

in hay ston

from get inde ();

toop surportion;

temp - took = NOLLS

solves tends

- territoria

#include < stdio.h> #include < stdlib.h> struct node

int info;
struct mode \*link;
};

typeder struct mode \*NODE;

```
NODE getrade()
    2 = (NODE) malloc (size of (struct node));
    NODE X;
    if (x = = NULL)
      prints (" Memory full (n");
     exit (0);
  return x;
roid freenade (NODE x)
  free(x);
NODE insert-node front (NODE first, int item)
   NODE temp;
    temp = getrode ();
    temp - info = item;
    temp - link = NULL;
   if (frist == NULL)
                       Norme: Krismoury
   return temp;
    temp - link = first;
                         LAB PROGRAM-8
    first = teng;
                        and to implement stack and
  setur fist;
NODE insert-rear (NODE first, int item)
   NODE temp, cur,
   temp = getrode ();
   temp -> info = item;
                                          ingu to
   temp - link = NULL;
   if (first == NULL)
     return temp;
  cur = first;
```

```
while ( cus - link ! = NULL)
  rus = cur -> link;
  eus - link = temp;
   neturn frist;
NODE delete-front (NODE frest)
 ? NODE temp;
   9 printf ("List is empty cannot delete \n");
   return first;
  temp = first;
  print (" Item deleted at front-end is = % d \n', first-sint)
  free (first);
  return temp;
NODE delete-rear (NODE first)
   NODE eur, prev;
    2 print ("List is empty cannot delete \n");
     return friet;
  y (first → link = = NULL)
                              %d\n", first → info }
 a printf (" Item deleted is
   free (first);
                           and the first stark for the same
    return NULL;
                          purely to temps the cover
  prev = NULL;
   cur = first;
```

```
while ( cur -> link ! = NULL)
     prer " reus;
   printy 1. Item deleted at rear end is % d"; cur singo);
    free (eur);
prev > link = NULL;
return first;
 roid display (NODE first)
   NODE temp;
   if (first == NULL)
     printf ("List is empty connot display items \n");
    return;
  print ("Contents of list: '\n");
  for (temp=first; temp!= NULL; temp= temp -> link)
    2 printf (" %d \n", temp -) info);
                                          NOOF LUG PER
                               there is trul " fring is with
roid main()
   int êtem, choice, pos, ê, n, court, key;
NODE pirst=NULL, a, b;
  for (33)
    printf (" In1: Stack In 2: Queux In 3: Fret In");
     print (" Enter the choice In");
     scary (" of-d", b choice);
                                               grand = And
    switch (choice)
                                               charge and
```

```
rease 1: printf ("stack \n");
    for (3 3)
   3 printf (" \n 1 . Inset_rear \n 2 . Delete - rear \n3: Display-
     list In 4° Exit In");
     printf ("Enter the chirce\n");
     scarf (" %d", & choice) =
     switch (choice)
   2 case 1° printf ("Enter the "item at reas-end \n");
scary ("%d", & item);
          first = pinsert-rear (first, item);
          break;
      case 2: first = delete-rear (first);
             break ;
        rase 30 display (first);
            break;
        default : exit (0);
              break;
   case 2: printf ("QUEUE \n");
       for (; ;)
        print (" In 1° Insert - rear In 2° Delete - front In
        3° Display - list \n 4° Exit \n");
      print ("Enter the choice (n");
      scanf (" % d", & choice);
      surtch (choice)
     case 1° printf ("Enter the "tem at rear-end \n");
               scanf (" olod", & item);
               first = insert-rear (first, item);
                break;
      case 2: first = delete-front (first);
                break;
```

of don't " thong I was case 3° display (first);
break; default: esuit (0); case 3° exit (0); défault : printf ("Invalid choice \n"); sees to print (. [ inter the ideas at sees and 12