```
p[repindex] = in (i);
  pafault and 19;
  displages ();
3 else
     brugg (, No bode tangi, );
 I also Pg Fault Cnot ();
 int main ()
    ent choice;
        print ("In Page Replacement Algorithms
In 1. Enter data In 2. FIFO in 3. optimalin 4. IRU
In 5. Exist In Enter your choice,");
 sans (" %d", & choice);
  Switch (choice)
  5 case 1: getpata ();
    Cose 2: 1001);
      break,
   Case 3: optimal ();
       break;
   case 4: 1-60();
        break;
   cas default: netwon b;
        books
```

OUTPUT Page Replacement Algorithms 1. Enten data 2. FIFO 3. Optimal 9. LPV 5. Exs+ Enter your charce:2 Enter length of page reference sequence: 12 Enter the page référence: 1234,251234, Enter the no of frames:3 For 2:12 For 1: 341 Por 2: 412 For 1: No page fault For 2: No page faut Por 3: 253 Fox 4: 534 For 5: No page fault Total no of page fauts: 9 Menu Enter your choice:5

a Write a CProgram to Stimulate disk Scheduling algorithms a) FCFS -> # include c. std:0. hs # Enclude - Std Ub.h > int main () { Pot RQ[100], i, n, Total HeadMoment = D, initial; prints (" Enter the number of Requests. (n"); Scans (" ofod", Sn); Prints ("Enter the Requests sequence \n"); for (120;141) Scant (no/od", 8 @RQ [i]); points ("Enter ontial head position \n"); 30anf (" 70d", LingHal); for (1201, 12n , 1++) PotalHeadMoment = TotalHead Moment + abs (RO(i) - Initial); initial = RU(i). prints (4 Total head moment is god" Tot at Head Moment); getino o; OUTPUT Enter the number of requests 5 Enter the requests sequence 5 98 107 45 78 Enter instial head position 45 notal Head Movement 95 237

b). SCAN # include 1stdio. h> # include 2 Std libh > Ent main () ? Int Rop[100], i, n, TotalHeadMoment = 0 Initial, Size, move; prints ("Enter the number of Requests In"); Scanf (" %d", &n); prints (" Finter the Requests sequence In"); tor (120°, 120°, 1++) Sconf (40/0d4, & RQ(i)); points ("Enter-Initial head position in"); scans ("4.d 4 85°ze); prints (" Enter the head movement direction for high I and for low o. In "); Scans (" nd 4, smove); 10x (1=0, 15 U, 1++) for (920°, 920-1-1-, 941) Py (RQ[;] > RQ[;+1]) int temp; temp= RQ(j) RQ(i) = RQ(j+1); ROSI +D 2 temp! 3

```
for (120; 12n; 1+1)
    of (initial < RQ [i])
      index = 1)
       book >
[] (move ==1)
[ for ( = Indon; 12n; 14+)
   2 Total Head Moment = Total Head Moment +
abs (R&(i) - initfal);
       Partfal = ROTI);
   Potal Head Moment = Potal Head Moment
abs(size - R&[:-1]-1);
 1 284°al = Size -1 5
 for 1 = ? rdex -1; 97 50; 1--)
       Potal Head Moment = Total Head Mament +
abs(RQ(:)-Pritfal);
   in Hal = R8 (1),
2 ols
{ for ( "12 Pindex -1; 1> > 0; 1--)
       Total Head Moment = notal Head Moment
 + abs (Ry [i] = Pno Hal);
    Pritial = RB (i);
abs (R91:+2-0)
```

Initial = 0; for 1= andox; PLn; P++? [Total Head Moment = Total Head Moment + abs (RQ(i) - Initoal); Initial = RQ(i); Protof " Total need movement Os % od" Total Head Moment); Deturn D; DUTPUT Firster the number of requests & Enter the original sequence 98 183 37 122 14 124 65 67 Enter initial head position Enter total olisk size 200 Enter the head movement direction for high, and for low o Potal Head Movement is 359

O C-SCAN # include coldio. h> # include (stallid.h) int main () Pot RQ(100), i, j. n., Totaltlead Mornent = 0, initial, size, move; points ("Enter the number of Requests In"); Scanf (" % d", 2n); prints (" Enter the Requests Sequence in "); por (120", 12n; 14) Scanf (4%du RROSI); points ("Brow initial head positioning); Sconflyod", 2 protod); posint of " Enter total disk size \n"); Scans (" "/d" & size); print of (" Enter the head movement direct" for high I and for low o'n"); Scand ("Tod", Smove); for (= 0, 12 n, 1++) for (j=0; j2n-1-1; j++) 3 9 (RO[1] > RO[3+1]) 2 gut tempi temp = Rafij; ROS[;] = ROS[;+D; ROCG+1] = temp 9

```
int index
  por (120° 960°, 144)
   if (mitial LRG(i))
      2 Andox = ";
          break;
  1 (move == 1)
  { for (is index; icn; it
      } Potally and Mannent = Potal Head Moment +
  abs [ RQ[i] - initial);
   gritial = Roli]:
Total Head Moment > Total Head Moment + abs ( Size
  - RO(1-1)-1);
Total Head Moment & Potal Head Moment + abs (Size-1-p);
  initial=0;
 for (120; 92 groder; 1+1)
     Total Head Moment + = abs (Reg [:] = 9,00 tial);
      initial = Rg(i);
9 else
{ por ( = 2 Proden - 1; 1520; = )
    2 Potal read Moment + = abs(RQ[i] - hitfal)
        inttial = Ry(i);
```

Total Head Movement + = abs (RQ[i+1]-0); Total Head Movement = Total Head Movement + abs (size -1-0); initial = Size -1; tor (is n-1:01) >= [nder : 0, --) { Potal Head Moment + = abs (Rg(i) - initial); Postal = Rosij prints (" Total head movement is " hod" Total Head Moment). return 03 OUTPUT Enter the number of requests 5 Frier the reguests sequence 98 183 37 122 19 Finter the inftfal Head posifion 53 Enter total disk size Enter the head monvement direction for high and for loss o Potal hoad Movement 95 359

15/5/24 FCFS and SIF 72 5 20 Pre-emptive & Round Robin Rate Monotonic & Forliest 5/6/24 Deadline First Producer & consumer, 12/6/24 Philosopher Dining Banker's algorithm, 19/6/24 Deadlock detection Contigous Memory allocation 317/24 Page replacement algorithm 10/7/24 Disk Scheduling