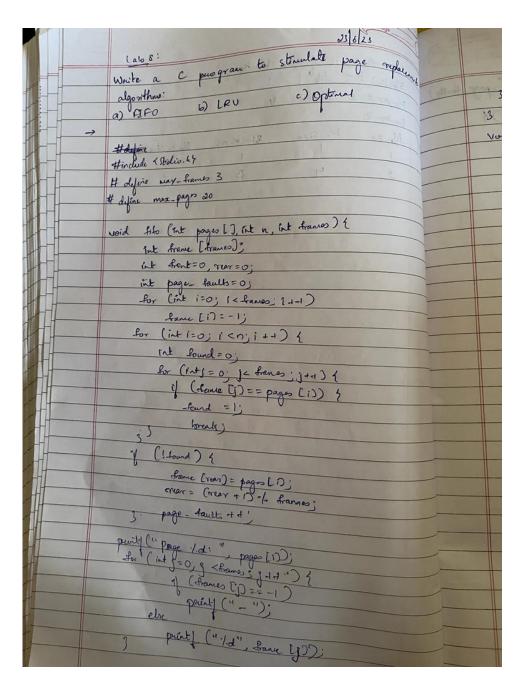
LAB9

Write a C program to simulate page replacement algorithms

- a) FIFO
- b) LRU
- c) Optimal



1544	
Brewerk	punt ("In");
2013	3 print ("In"); print ("Total page faults (FIFO): "I'd In", page faults);
	3 Land the or and Oliver the leading had
IA .	a l' Ciet mean [] int in int flames)?
	Ent france (frances);
	for (mit i=0; ix frames); for (mit i=0; ix frames) id+);
	int used [max-pages] = lobj
	for (mt 170) ix frames) id+1)
	Faul (i) = - (j)
	lo (ak i=0; 1kn jir +n) (in)
	tound = 0)
	Por (int 1=0; 1< frames ; j++) {
	gl (frame [j)== pages (1)) 1
	found = 1;
	q (frame (j) = = pages (i)) 1 found = 1; used (j) = 1;
	brakes
	brake
	i) (1-found) {
	if (1-found) { int min=0;
	for (int 1=1; 1 x frames; 1++) 1
	of Cused [] < used (mid))
	salar van min's 1 di
	t (and the state of day) at
	frame [min) = pages [i];
	used truin) = 13-
	page laults + + ;
	pounty ("Page Id: " pages (i));
	for City 12011 & Paris
100	pourly ("Page 1.d: ", pages (17); for (int j=0; j < frames ; j+1) { 7 (frame (1) = -, -1)
	1 Charle () 3 - 3 - 2)
	paint ("_");
	em
	point ("lid", fram (1))
	pount ("I.d", frame (j))

	Pulls (LRV): Ird In , page o
	puit ("Total page faults (LRV): /id In , page Soults) Void optimal (int pages [], int n, int frames)? Void optimal (int pages [], int n, int frames)? Int frame [Aramo] For (int 1=0; i < frames); i =1-1) Frame [i) = -1; Int found = 0; Int (int 1=0; i < frames); i =1+1)?
((3) ust	1 1 Cox mans [] int n, int frames) 5
	Void optimal (Inc framo)
	it faults = 09
	In (int 150; ic famos i +1+1)
	frame (i) = -1;
	for (inter; iso int)?
	int found = 0 3 (1) and
	for (int jeo) je franco; j. ++) {
	1 (fame [j] == pages [i]) ?
	bund = 1;
	((nt 50) 2 (rans) 147 (fame [] = pages [1]) { bund = 1; break; 3 (! found) ?
	of Clared 25
	11 (1 (10 1)
	france [i] = page [i]:
	else 3 (Carolina)
3	int max dist - 1.
	If (! found)? If (! frames) Arrange [i] = pages [i]; else ? int mass dist = -1; fint aeplace - page =-1; For[ent]=0; Frames; + +) { bit dist = MAX - PAGES.
	Sollent 1-0; 1 Frances: 11+24
	for (int 15:11)
	for (int K=1+1: K<0: K+1) }
	1 (pages [K] == france (1)) 4
	for (int K=i+1: K <n; (j))="" (pages="" 1)="" [k]="=" break;<="" chit="K-3;" france="" k+1)="" td="" {=""></n;>
	break;
	71 (4:1)
	(chuye) man_dist) {
	1) (duit > man_dist) { max_duit = dist, mplace page = j
	3 space page = 1
	3 frame broken
	J Page - lauts : 1, Page J = pages [D.
	3 franc Implace - page) = pages [1];

punt 1 ("Page 1 d" pages [1]):

for (int j=0' j < france j++) ?

(franc j)==-1)

punt ("-"):

else - faulty). 3 punt ("hd", franc [j]) Durt ("Total page faults Coptinal): 1. of In ", page - faults) Int main () 4 int pages [max-pages]; puint ("Exter the number of pages:");
sear ("1.d", 2n);
puint ("Enter the references string; ");
lor (int i=0; i=n; 1+1) scar ("'I'd" & pages (i));

puilty ("Entor the no. of frames; 1');

scar ("'I'd" & frames); punt (" Enter the reference story: ")'
for ("int 1-0; 1<0; 1+1) grand ("!d", & pages [i]),

printly ("Ento the number of frames:");

grant ("ld", & frames; ");

printly ("Ento the number of frames; "); dean (" A.d", & frames); / points ("Enter the number of frances: ");

Scan ("1.d", & frances);

pount) ("Enter Filto page Replacement: "");

Plyo (pages, n frances);

pount) ("LRV Page sephement: "");

Iru (pages, n, Frances);

point) ("Optimal Page Replacement: "");

optimal (pages pr. hanes); Output: Enter the number of pages: 14

Enter page no s 10 4 3 21 4 6 3 c

3 8 3 Enter the no. of frames: 3 Optimal · page fault = 18

Optimal · page fault = 16 page fault = 18

LRU - page lault = 18 (A) (A) (A) (A) (A) (A) " i prote amorper all return ") (3) (C01) V Dec 16 61 10

OUTPUT:

Press any key to continue.

"C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\OS\bin\Debug\OS.exe" Enter Size of memory: Enter number of process in queue: Enter 6 process 7 4 10 4 2 1 FIFO: Memory: 700 Memory: 7 4 0 Memory: 7 4 10 Memory: 7 4 10 Memory: 1 4 10 LRU: Memory: 700 Memory: 7 4 0 Memory: 7 4 10 Memory: 7 4 10 Memory: 7 4 1 Optimal: Memory: 7 0 0 Memory: 7 4 0 Memory: 7 4 10 Memory: 7 4 10 Memory: 1 4 10 Process returned 6 (0x6) execution time: 14.298 s