7)Develop a C program to simulate page replacement algorithms: a) FIFO b) LRU

a)FIFO

#include <stdio.h>

#include <stdlib.h>

int fr[3];

void main()

{

void display();

int i, j, page[12] = {2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2};

int flag1 = 0, flag2 = 0, pf = 0, frsize = 3, top = 0;

for (i = 0; i < 3; i++)

{

fr[i] = -1;

}

for (j = 0; j < 12; j++)

{

flag1 = 0;

flag2 = 0;

for (i = 0; i < 12; i++)

{

if (fr[i] == page[j])

{

flag1 = 1;

flag2 = 1;

break;

}

}

if (flag1 == 0)

{

for (i = 0; i < frsize; i++)

{

if (fr[i] == -1)

{

fr[i] = page[j];

flag2 = 1;

break;

}

}

}

if (flag2 == 0)

{

fr[top] = page[j];

top++;

pf++;

if (top >= frsize)

top = 0;

}

display();

}

printf("Number of page faults : %d ", pf + frsize);

exit(0);

}

void display()

{

int i;

printf("\n");

for (i = 0; i < 3; i++)

printf("%d\t", fr[i]);

}

**b)LRU**

#include <stdio.h>

#include <stdlib.h>

int fr[3];

void main()

{

void display();

int p[12] = {2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2}, i, j, fs[3];

int index, k, l, flag1 = 0, flag2 = 0, pf = 0, frsize = 3;

for (i = 0; i < 3; i++)

{

fr[i] = -1;

}

for (j = 0; j < 12; j++)

{

flag1 = 0, flag2 = 0;

for (i = 0; i < 3; i++)

{

if (fr[i] == p[j])

{

flag1 = 1;

flag2 = 1;

break;

}

}

if (flag1 == 0)

{

for (i = 0; i < 3; i++)

{

if (fr[i] == -1)

{

fr[i] = p[j];

flag2 = 1;

break;

}

}

}

if (flag2 == 0)

{

for (i = 0; i < 3; i++)

fs[i] = 0;

for (k = j - 1, l = 1; l <= frsize - 1; l++, k--)

{

for (i = 0; i < 3; i++)

{

if (fr[i] == p[k])

fs[i] = 1;

}

}

for (i = 0; i < 3; i++)

{

if (fs[i] == 0)

index = i;

}

fr[index] = p[j];

pf++;

}

display();

}

printf("\n no of page faults :%d", pf + frsize);

exit(0);

}

void display()

{

int i;

printf("\n");

for (i = 0; i < 3; i++)

printf("\t%d", fr[i]); }