FIFO

#include <stdio.h>

#define MAX\_FRAMES 3

int main() {

int frames[MAX\_FRAMES], page\_faults = 0, num\_frames = 0, num\_pages = 8;

int page\_references[] = {4, 1, 2, 4, 3, 2, 1, 5};

int i, j, k;

// Initialize all frames to -1 to indicate that they are empty

for (i = 0; i < MAX\_FRAMES; i++) {

frames[i] = -1;

}

// Iterate through the page reference sequence

for (i = 0; i < num\_pages; i++) {

int page = page\_references[i];

int page\_found = 0;

// Check if the page is already in a frame

for (j = 0; j < MAX\_FRAMES; j++) {

if (frames[j] == page) {

page\_found = 1;

break;

}

}

// If the page is not in a frame, replace the oldest frame with the new page

if (!page\_found) {

frames[num\_frames % MAX\_FRAMES] = page;

num\_frames++;

page\_faults++;

}

// Print the current state of the frames

printf("Page %d: [", page);

for (k = 0; k < MAX\_FRAMES; k++) {

if (frames[k] == -1) {

printf(" ");

} else {

printf("%d", frames[k]);

}

if (k < MAX\_FRAMES - 1) {

printf("|");

}

}

printf("]\n");

}

// Print the total number of page faults

printf("Total page faults: %d\n", page\_faults);

return 0;

}

OUT PUT:

Page 4: [4| | ]

Page 1: [4|1| ]

Page 2: [4|1|2]

Page 4: [4|1|2]

Page 3: [3|1|2]

Page 2: [3|1|2]

Page 1: [3|1|2]

Page 5: [3|5|2]

Total page faults: 5

--------------------------------

Process exited after 0.05113 seconds with return value 0

Press any key to continue . . .