FIFO --

#include<stdio.h>

int main(){

int frames, pages, i, j, hit=0, fault=0, counter=0;

int reference\_string[100], mem\_layout[100][100];

printf("\nEnter the number of frames: ");

scanf("%d",&frames);

printf("\nEnter the number of pages: ");

scanf("%d",&pages);

printf("\nEnter the reference string: ");

for(i=0;i<pages;i++){

scanf("%d",&reference\_string[i]);

}

for(i=0;i<frames;i++){

mem\_layout[i][0]=-1;

}

for(i=0;i<pages;i++){

hit=0;

for(j=0;j<frames;j++){

if(mem\_layout[j][0]==reference\_string[i]){

hit=1;

mem\_layout[j][i+1]=1;

break;

}

}

if(hit==0){

mem\_layout[counter][0]=reference\_string[i];

fault++;

for(j=0;j<frames;j++){

mem\_layout[counter][j+1]=0;

}

counter++;

if(counter==frames){

counter=0;

}

}

printf("\n");

for(j=0;j<frames;j++){

printf("%d\t",mem\_layout[j][i]);

}

}

printf("\nTotal Page Faults: %d",fault);

return 0;

}

LRU--

#include <stdio.h>

#define MAX\_PAGES 100

int findLRU(int time[], int n){

int i, min = time[0], pos = 0;

for(i=1;i<n;i++){

if(time[i]<min){

min = time[i];

pos = i;

}

}

return pos;

}

int main(){

int pages[MAX\_PAGES], frames, n, i, j, k, faults = 0, pos;

printf("Enter the number of frames: ");

scanf("%d", &frames);

printf("Enter the number of pages: ");

scanf("%d", &n);

printf("Enter the reference string: ");

for(i=0;i<n;i++){

scanf("%d", &pages[i]);

}

int mem[frames], time[frames];

for(i=0;i<frames;i++){

mem[i] = -1;

time[i] = 0;

}

for(i=0;i<n;i++){

for(j=0;j<frames;j++){

if(mem[j]==pages[i]){

time[j] = i+1;

break;

}

}

if(j==frames){

pos = findLRU(time, frames);

mem[pos] = pages[i];

time[pos] = i+1;

faults++;

}

printf("\n");

for(k=0;k<frames;k++){

printf("%d\t", mem[k]);

}

}

printf("\nTotal Page Faults: %d", faults);

return 0;

}

OPTIMAL--

#include<stdio.h>

#include<limits.h>

int main(){

int frames, pages, i, j, k, l, hit=0, fault=0, max\_dist, max\_frame, flag;

int reference\_string[100], mem\_layout[100][100], distance[100];

printf("\nEnter the number of frames: ");

scanf("%d",&frames);

printf("\nEnter the number of pages: ");

scanf("%d",&pages);

printf("\nEnter the reference string: ");

for(i=0;i<pages;i++){

scanf("%d",&reference\_string[i]);

}

for(i=0;i<frames;i++){

mem\_layout[i][0]=-1;

}

for(i=0;i<pages;i++){

hit=0;

for(j=0;j<frames;j++){

if(mem\_layout[j][0]==reference\_string[i]){

hit=1;

break;

}

}

if(hit==0){

fault++;

flag=0;

for(j=0;j<frames;j++){

if(mem\_layout[j][0]==-1){

mem\_layout[j][0]=reference\_string[i];

flag=1;

break;

}

}

if(flag==0){

for(j=0;j<frames;j++){

distance[j]=INT\_MAX;

for(k=i+1;k<pages;k++){

if(reference\_string[k]==mem\_layout[j][0]){

distance[j]=k-i;

break;

}

}

}

max\_dist=-1;

for(j=0;j<frames;j++){

if(distance[j]>max\_dist){

max\_dist=distance[j];

max\_frame=j;

}

}

mem\_layout[max\_frame][0]=reference\_string[i];

}

}

for(j=0;j<frames;j++){

mem\_layout[j][i+1]=mem\_layout[j][i];

}

}

printf("\nTotal Page Faults: %d",fault);

return 0;

}