**PAGE REPLACEMENTS ALGORITHM**

**FIFO , LRU , LFU**

SANJANA S NAIR

CSE C

33

#include <stdio.h>

#define MAX 100

int count[MAX], time[MAX], inframe[MAX], q[200];

void fifo(int a[], int n, int frames)

{

int miss = 0, i, c = 0, j = 0;

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

{

if (inframe[a[i]] == 0)

{

if (c < frames)

{

q[c++] = a[i];

}

else

{

inframe[q[j]] = 0;

q[j] = a[i];

j = (j + 1) % frames;

}

inframe[a[i]] = 1;

printf("Frames: ");

for (int k = 0; k < c; k++)

{

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

}

printf("\n");

miss++;

}

}

printf("Page faults = %d\n", miss);

}

void lru(int a[], int n, int frames)

{

int miss = 0, c = 0, loc, min, i, j;

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

{

time[a[i]] = i;

if (inframe[a[i]] == 0)

{

if (c < frames)

{

q[c++] = a[i];

}

else

{

for (j = 1, loc = 0, min = time[q[0]]; j < frames; j++)

{

if ((time[q[j]]) < min)

{

min = time[q[j]];

loc = j;

}

}

inframe[q[loc]] = 0;

q[loc] = a[i];

}

inframe[a[i]] = 1;

printf("Frames: ");

for (int k = 0; k < c; k++)

{

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

}

printf("\n");

miss++;

}

}

printf("Page faults = %d\n", miss);

}

void lfu(int a[], int n, int frames)

{

int miss = 0, c = 0, loc, min, i, j;

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

{

count[a[i]]++;

if (inframe[a[i]] == 0)

{

if (c < frames)

{

q[c++] = a[i];

}

else

{

for (j = 1, loc = 0, min = count[q[0]]; j < frames; j++)

{

if ((count[q[j]]) < min)

{

min = count[q[j]];

loc = j;

}

}

inframe[q[loc]] = 0;

q[loc] = a[i];

}

inframe[a[i]] = 1;

printf("Frames: ");

for (int k = 0; k < c; k++)

{

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

}

printf("\n");

miss++;

}

}

printf("Page faults = %d\n", miss);

}

int main()

{

int i, n, a[MAX], frames, ch;

printf("Enter the number of references in reference string:");

scanf("%d", &n);

printf("Enter reference string:\n");

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

{

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

}

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

scanf("%d", &frames);

do

{

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

{

count[a[i]] = 0;

time[a[i]] = 0;

inframe[a[i]] = 0;

}

printf("Enter 1:FIFO 2:LRU 3:LFU 4:Exit:\n");

scanf("%d", &ch);

switch (ch)

{

case 1: fifo(a, n, frames);

break;

case 2: lru(a, n, frames);

break;

case 3: lfu(a, n, frames);

break;

case 4: break;

default: printf("Invalid input\n");

}

} while (ch!= 4);

return 0;

}

**SAMPLE OUTPUT**

Enter the number of references in reference string: 10

Enter reference string:

7

0

1

2

3

4

0

3

2

1

Enter the number of frames: 3

Enter 1:FIFO 2:LRU 3:LFU 4:Exit:

1

Frames: 7

Frames: 7 0

Frames: 7 0 1

Frames: 2 0 1

Frames: 2 3 1

Frames: 2 3 4

Frames: 0 3 4

Frames: 0 2 4

Frames: 0 2 1

Page faults = 9

Enter 1:FIFO 2:LRU 3:LFU 4:Exit:

2

Frames: 7

Frames: 7 0

Frames: 7 0 1

Frames: 2 0 1

Frames: 2 3 1

Frames: 2 3 4

Frames: 0 3 4

Frames: 0 3 2

Frames: 1 3 2

Page faults = 9

Enter 1:FIFO 2:LRU 3:LFU 4:Exit:

3

Frames: 7

Frames: 7 0

Frames: 7 0 1

Frames: 2 0 1

Frames: 3 0 1

Frames: 4 0 1

Frames: 3 0 1

Frames: 3 0 2

Frames: 1 0 2

Page faults = 9

Enter 1:FIFO 2:LRU 3:LFU 4:Exit:

4