实验2.3页面置换

#include <stdio.h>

#include <signal.h>

#include <unistd.h>

#include<stdlib.h>

#include<string.h>

#include<sys/wait.h>

#include <sys/types.h>

#include <time.h>

#define max 100000

typedef struct{ // 1 2 3 4 1 2 5 1 2 3 4 5/12 个数据

int pos; // 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1/20 个数

int valid;

int latest;

}S;

int main()

{

int total\_instruciton=0; //页面数量

const int ap=20; //内存分配页面数,随机生成的页面大小在 0 到 ap-1之间

const int pp=3; //进程页面数

int dise=0;

int front=0,rear=0,a;

int order[max];

S page[ap];

S pc[pp+1];

memset(page,0,sizeof(page));

memset(pc,0,sizeof(pc));

printf("手工输入请输入 1,随机输入请输入 0\n");

scanf("%d",&a);

if(a) //1 表示手工输入页面序列，0 表示随机输入页面序列

{

printf("回车表示输入结束\n");

scanf("%d",&order[0]);

for(int i=1;getchar()!='\n';i++) //手工输入，回车作为结束符

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

total\_instruciton++;

}

}

else {

printf("请输入随机生成页面的个数：");//定义随机生成的数组个数

scanf("%d",&total\_instruciton);

srand((unsigned int)time(NULL));

for(int i=0;i<total\_instruciton;i++) //随机输入

{

order[i]=rand()%ap;

printf("%d ",order[i]);

}

printf("\n");

}

printf("FIFO:\n");

for(int i=0;i<total\_instruciton;i++)

{

if(page[order[i]].valid==0)

if(front==(rear+1)%(pp+1))

{

pc[rear].pos=order[i];

pc[rear].valid=1;

page[order[i]].pos=rear;

page[order[i]].valid=1;

rear=(rear+1)%(pp+1);

pc[front].valid=0;

page[pc[front].pos].valid=0;

front=(front+1)%(pp+1);

dise++;

}

else {

pc[rear].pos=order[i];

pc[rear].valid=1;

page[order[i]].pos=rear;

page[order[i]].valid=1;

rear=(rear+1)%(pp+1);

dise++;

}

for(int j=front;j!=rear;j=(j+1)%(pp+1))

printf("%d ",pc[j].pos);

printf("\n");

}

printf("FIFO 的缺页率为:%lf\n",(double)dise/total\_instruciton);

memset(page,0,sizeof(page));

memset(pc,0,sizeof(pc));

dise=0;

int count=0;

printf("LRU:\n");

for(int i=0;i<total\_instruciton;i++)

{

if(page[order[i]].valid==0)

{

dise++;

if(count<pp)

{

page[order[i]].pos=count;

page[order[i]].valid=1;

pc[count].pos=order[i];

pc[count].valid=1;

count++;

}

else{

int temp=0x7fffffff,k;

for(int j=0;j<pp;j++)

if(pc[j].latest<temp)

{

temp= pc[j].latest;

k=j;

}

page[pc[k].pos].valid=0;

pc[k].valid=1;

pc[k].pos=order[i];

page[order[i]].valid=1;

page[order[i]].pos=k;

}

}

pc[page[order[i]].pos].latest=i;

page[order[i]].latest=i;

for(int j=0;j<pp&&pc[j].valid==1;j++)

printf("%d ",pc[j].pos);

printf("\n");

}

printf("LRU 的缺页率为:%lf\n",(double)dise/total\_instruciton);

}