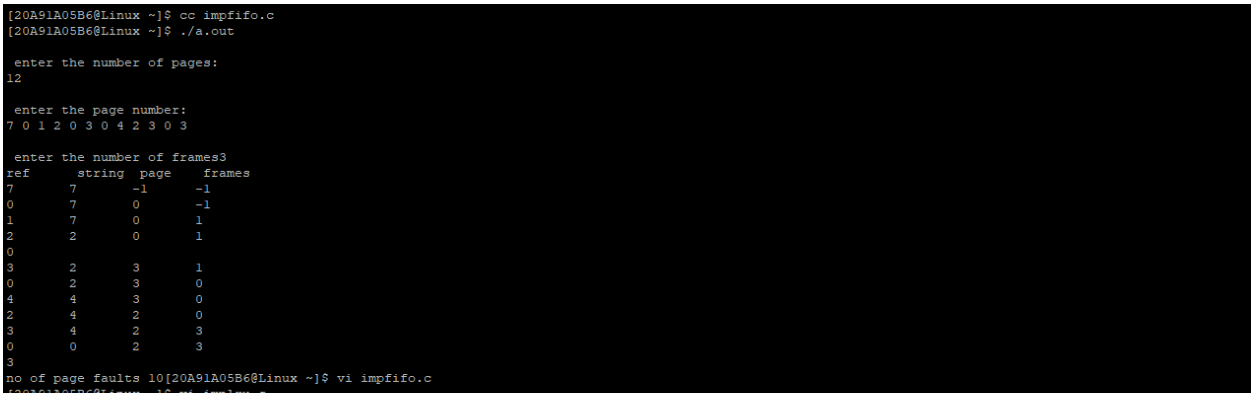
7. Simulate the FIFO page replacement algorithm

AIM:TO IMPLEMENT FIFO PAGE REPLACEMENT PROGRAM

Program:

```
 #include<stdio.h>

int main()

{

int i,j,n,a[50],frame[10],no,k,avail,count=0;

printf("\n enter the number of pages:\n");

scanf("%d",&n);

printf("\n enter the page number:\n");

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

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

printf("\n enter the number of frames");

scanf("%d",&no);

for(i=0;i<no;i++)

frame[i]=-1;

j=0;

printf("ref\t string\t page\t frames\n");

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

{

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

avail=0;
for(k=0;k<no;k++)

if(frame[k]==a[i])

avail=1;

if(avail==0)

{

frame[j]=a[i];

j=(j+1)%no;

count++;

for(k=0;k<no;k++)

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

}

printf("\n");

}

printf("no of page faults %d",count);

return 0;

}
```

Output:



```
[20A91A05B6@Linux ~]$ cc impfifo.c
[20A91A05B6@Linux ~]$ ./a.out

 enter the number of pages:
12

 enter the page number:
7 0 1 2 0 3 0 4 2 3 0 3

 enter the number of frames3
ref      string  page    frames
7       7       -1      -1
0       7       0       -1
1       7       0       1
2       2       0       1
0
3       2       3       1
0       2       3       0
4       4       3       0
2       4       2       0
3       4       2       3
0       0       2       3
3
no of page faults 10[20A91A05B6@Linux ~]$ vi impfifo.c
```

8. Simulate the LRU page replacement algorithm

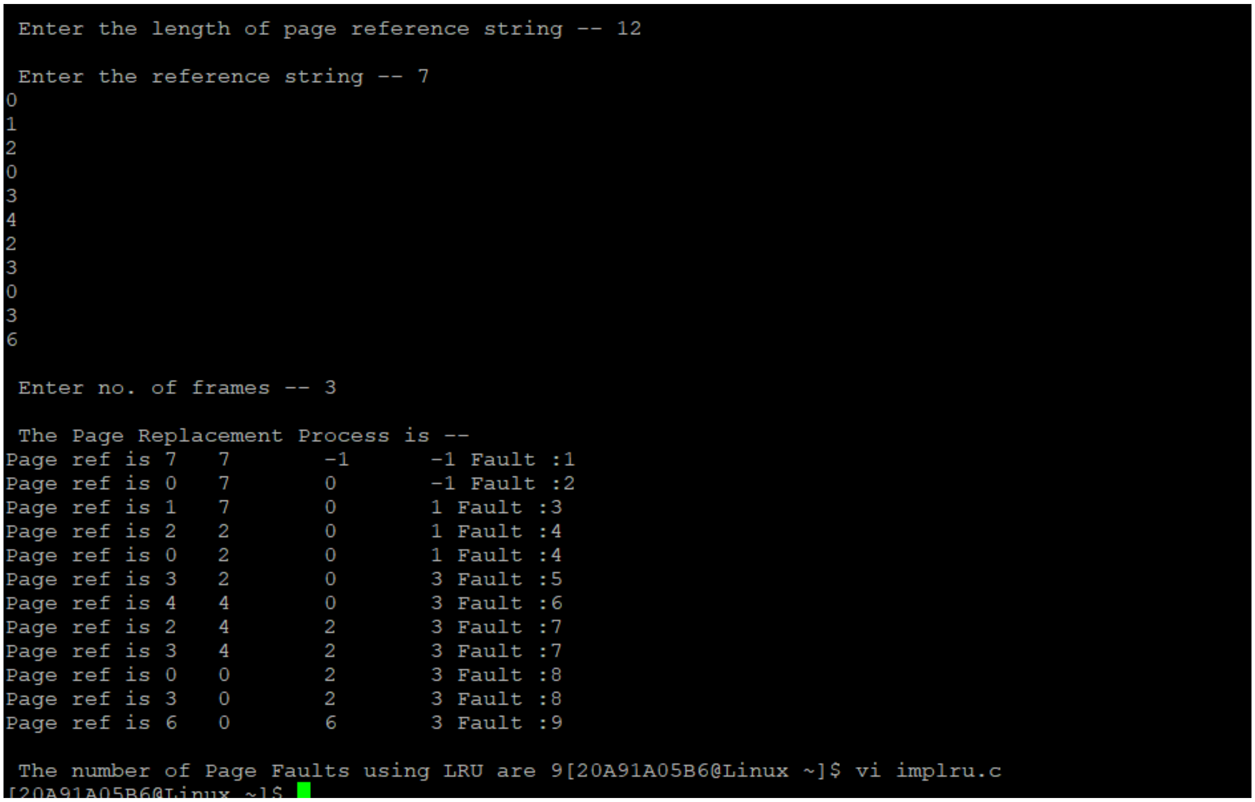
AIM:TO IMPLEMENT LRU PAGE REPLACEMENT PROGRAM

Program:

```
#include <stdio.h>
int main()
{
int i, j, k, f,max,p=10, pf=0, count[10], pageref[25], fp[10], n,flag[10];
printf("\n Enter the length of page reference string -- "); scanf("%d",&n);
printf("\n Enter the reference string -- ");
for(i=0;i<n;i++)
        scanf("%d",&pageref[i]);
printf("\n Enter no. of frames -- ");
scanf("%d",&f);
for(i=0;i<f;i++)
 {
        fp[i]=-1;count[i]=0;flag[i]=0;
}
printf("\n The Page Replacement Process is -- \n");
for(i=0;i<n;i++)
{
        for(k=0;k<f;k++)
        {
        if(count[k]==0)
        {
                fp[k]=pageref[i];
                pf++;
                count[k]=1;p=k;flag[k]=1; break;
        }
        else if(fp[k]==pageref[i])     //required page found
        {

                count[k]=1;p=k;flag[k]=1; break;
        }
        }
```

```
    if(k==f) //LRU replacement
    {
        max=0;
        for(j=0;j<f;j++)
        {
            if( count[j]>max)
            {
                max=count[j];
                p=j;
            }
        }
        fp[p]=pageref[i];
        count[p]=1;
        flag[p]=1;
        pf++;
    }
    printf("Page ref is %d",pageref[i]);
    for(j=0;j<f;j++)
    {
if(j==p || count[j]==0)
continue;
count[j]=count[j]+1;
}
    for(j=0;j<f;j++)
    {
        printf("\t%d ",fp[j]);
    }
        printf("Fault :%d",pf);
        printf("\n");
    }
printf("\n The number of Page Faults using LRU are %d",pf);
}
```

output:



```
 Enter the length of page reference string -- 12

 Enter the reference string -- 7
0
1
2
0
3
4
2
3
0
3
6

 Enter no. of frames -- 3

 The Page Replacement Process is --
Page ref is 7   7       -1      -1 Fault :1
Page ref is 0   7       0       -1 Fault :2
Page ref is 1   7       0       1 Fault :3
Page ref is 2   2       0       1 Fault :4
Page ref is 0   2       0       1 Fault :4
Page ref is 3   2       0       3 Fault :5
Page ref is 4   4       0       3 Fault :6
Page ref is 2   4       2       3 Fault :7
Page ref is 3   4       2       3 Fault :7
Page ref is 0   0       2       3 Fault :8
Page ref is 3   0       2       3 Fault :8
Page ref is 6   0       6       3 Fault :9

 The number of Page Faults using LRU are 9[20A91A05B6@Linux ~]$ vi implru.c
[20A91A05B6@Linux ~]$
```