**Q: 22**

**Program to implement FIFO page replacement algorithm .First generate a random –page-reference string where page ranges from 0 to 9.**

**We need to apply random page-reference string to each algorithm , and the we need to record the number of page fault incurred by each algorithm .**

**In this problem in order to generate 0-9 pages we should use page –replacement-algorithm .. And we should use page demand so that we can generate 1 -7 number of frame pages ..**

**CODE:**

#include<stdio.h>

#include<string.h>

#include<conio.h>

#include<stdlib.h>

#include<time.h>//for manipulating time

Int Random\_check=0;

Typedef long long ll;

Void Inputvalues(ll\*virtual\_pages,ll total\_pages, ll total\_frames);

Int pagehitting(ll \*frames\_list, ll total\_frames , ll goals);

{

For(int i=0; i<total\_frames; i++)

{

If(frames\_list[i]==goal)

Return 1;

}

Return 0;

}

Void genertepages(ll \*virtual\_pages, ll total\_pages)

{

For(int i=0; i<total\_pages; i++)

Virtual\_pages[i]==(rand()%10)//for generation of random 0-9 pages

}

Int main(int m, const , char\*\*n)

{

Srand(time(NULL));//internal clock to control choice of seed

ll total\_pages;

ll\*virtual\_pages;

if(m==1)

{

Printf("Enter number of virtual pages");

Scanf("%lld",&total\_pages);

Virtual\_pages=(ll\*)malloc(sizeof(ll))\*total\_pages);

Int generate\_pages(virtual\_pages,total\_pages);

Random\_check=1;

}

Else

{

Total\_page=m-1;

Virtual\_pages=(ll\*)malloc(sizeof(ll)\*total\_pages);

Int index=0;

For(int i=1;i<m;i++)

Virtual\_pages[index++]=atoll(n[i]);

}

ll total\_frames;

printf("Enter total number of frames");

scanf("%lld,&total\_frames");

inputs(virtual\_pages, total\_pages, total\_frames);

ll \*frames\_list=(ll\*)malloc(total\_frames\*sizeof(ll));

memoryset(frames\_list,-1,total\_frames\*,sizeof(ll));

ll count\_page\_hitting;

count\_page\_fault;

count\_page\_hitting=count\_page\_fault=0;

ll last=-1;

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

{

If(get pagehit(frames\_list,total\_frames,virtual\_pages[i]))

Count\_page\_hitting++;

Else

{

Frames\_list[++last%total\_frames]=virtual\_pages[i];

Count\_page\_fault++;

}

}

Printf("number of page hits are%lld\n",count\_page\_hitting);

Printf("number of page faults are %lld\n",count\_page\_fault);

Return 0;

}

Void inputs(ll\* virtual\_pages, ll total\_pages, ll total\_frames)

{

Printf("Entered data as follows");

Printf("Number of pages : %lld\n", total\_pages);

If(random\_check)

Printf("Random");

Printf("virtual pages");

For(int i=0;i<total\_pages;i++){

Printf("%dlld",virtual\_pages[i]);

Printf("number of frames ",total\_frames);

}

}