**Collage : Vishwakarma Institute of Technology**

**Course Name : Data Structure in C**

**Name : Vedika Vikas Sontakke**

**Roll no : 37**

**PRN NO 12220206**

**Assignment 3 : Page replacement algorithms**

1. **First Come First Serve**
2. **Least Recently Used**
3. **Optimal page replacement algorithm**

**Program : First Come First Serve**

/\*

ENTER THE NUMBER OF PAGES:  20

ENTER THE PAGE NUMBER :       7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

ENTER THE NUMBER OF FRAMES :3

\*/

#include<stdio.h>

int main()

{

    int no\_pages , no\_frames ;

    printf("enter number of pages : ");

    scanf("%d",&no\_pages);

    printf("enter number of frames : ");

    scanf("%d",&no\_frames);

    int pages[no\_pages] , frames[no\_frames];

    int current\_frame = 0 , page\_fault = 0 , is\_valid;

    printf("pages are : ");

    for(int i=0 ; i<no\_pages ; i++)

       scanf("%d",&pages[i]);

    for(int i=0 ; i<no\_frames ; i++)

       frames[i] = -1;

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

    for(int i=0 ; i<no\_pages ; i++)

    {

        printf("%d\t\t",pages[i]);

         is\_valid = 0;

         for(int k=0 ; k<no\_frames ; k++)

         {

            if(pages[i] == frames[k])

            {

                is\_valid = 1;

                break;

            }

         }

            if(is\_valid == 0)

            {

                frames[current\_frame] = pages[i];

                current\_frame = (current\_frame+1)%no\_frames;

                page\_fault++;

            }

            for(int l=0 ; l<no\_frames ; l++)

            {

                printf("%d\t",frames[l]);

            }

            printf("\n");

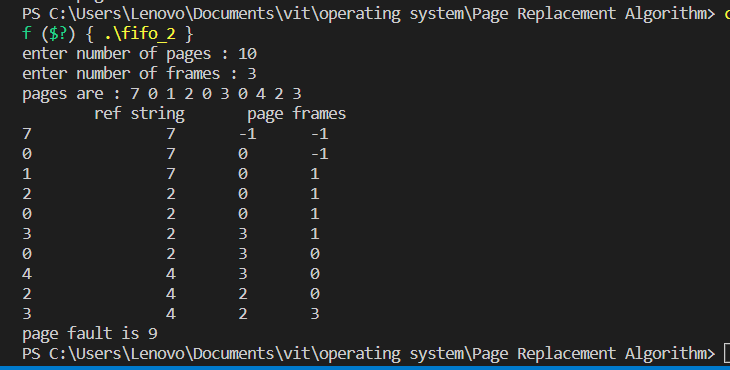
    }

     printf("page fault is %d ", page\_fault);

     return 0;

}

**Output :**

****

**Program : Least Recently Used**

/\*

Enter no of pages:10

Enter the reference string:7 5 9 4 3 7 9 6 2 1

Enter no of frames:3

\*/

#include<stdio.h>

int findLRU(int time[] , int no\_frames)

{

    int minium = time[0] , pos = 0 ;

    for(int i=1 ; i<no\_frames ; i++)

    {

          if(time[i] < minium)

          {

             minium = time[i];

             pos = i;

          }

    }

    return pos;

}

int main()

{

    int no\_pages , no\_frames ;

    printf("enter number of pages : ");

    scanf("%d",&no\_pages);

    printf("enter number of frames : ");

    scanf("%d",&no\_frames);

    int pages[no\_pages] , frames[no\_frames] , time[no\_frames] , counter=0 , faults=0 , pos;

    printf("enter referance string : \n");

    for(int i=0 ; i<no\_pages ; i++)

    {

        scanf("%d",&pages[i]);

    }

    for(int i=0 ; i<no\_frames ; i++)

    {

        frames[i] = -1 ;

    }

        printf("page \t f1 \t\t f2 \t\t f3 \t\t \n");

    //outer loop

    for(int i=0 ; i<no\_pages ; i++)

    {

        int flag1 = 0 , flag2 = 0;

        // if page presnet

        for(int j=0 ; j<no\_frames ; j++)

        {

            if(pages[i] == frames[j])

            {

                counter++;

                faults++;

                time[j] = counter;

                flag1 = 1;

                flag2 = 1;

                break;

            }

        }

        // if frame is free

        if(flag1 == 0)

       {

            for(int j=0 ; j<no\_frames ; j++)

        {

            if(frames[j] == -1)

            {

                counter++;

                faults++;

                frames[j] = pages[i];

                time[j] = counter;

                flag2 = 1;

                break;

            }

        }

       }

        // if frame is not free

        if(flag2 == 0)

        {

            pos = findLRU(time , no\_frames);

            counter++;

            faults++;

            time[pos] = counter;

            frames[pos] = pages[i];

        }

        printf("%d \t ",pages[i]);

        for(int j=0 ; j<no\_frames ; j++)

        {

            printf("%d \t \t",frames[j]);

        }

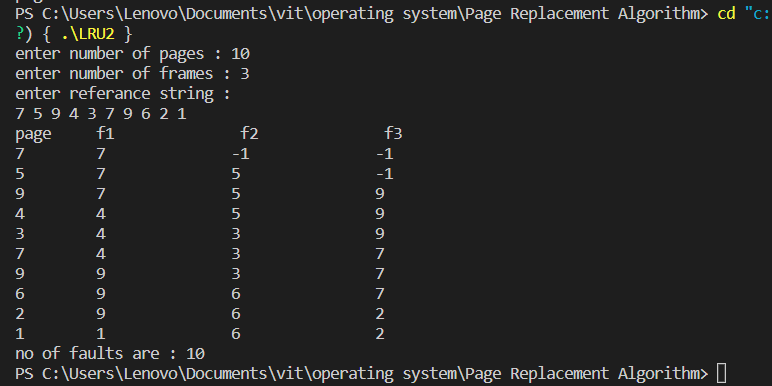
        printf("\n");

    }

    printf("no of faults are : %d \n",faults);

}

**Output :**

****

**Program : Optimal Page replacement Algorithm**

/\*

Enter number of frames: 3

Enter number of pages: 10

Enter page reference string: 2 3 4 2 1 3 7 5 4 3

\*/

#include<stdio.h>

int main()

{

      int no\_pages , no\_frames ;

    printf("enter number of pages : ");

    scanf("%d",&no\_pages);

    printf("enter number of frames : ");

    scanf("%d",&no\_frames);

    int pages[no\_pages] , frames[no\_frames] , temp[no\_frames] , faults=0 , pos , flag3;

    printf("enter referance string : \n");

    for(int i=0 ; i<no\_pages ; i++)

    {

        scanf("%d",&pages[i]);

    }

    for(int i=0 ; i<no\_frames ; i++)

    {

        frames[i] = -1 ;

    }

        printf("page \t f1 \t\t f2 \t\t f3 \t\t \n");

        //outer loop

        for(int i=0 ; i<no\_pages ; i++)

        {

            int flag1 = 0;

            int flag2 = 0;

            // if page found in frame

            for(int j=0 ; j<no\_frames ; j++)

            {

                if(frames[j] == pages[i])

                {

                    flag1 = 1;

                    flag2 = 1;

                    break;

                }

            }

            // if frame is free

            if(flag1 == 0)

            {

                for(int j=0 ; j<no\_frames ; j++)

               {

                   if(frames[j] == -1)

                   {

                       frames[j] = pages[i];

                       faults++;

                       flag2 = 1;

                       break;

                    }

                }

            }

            //temp how many times page found

            if(flag2 == 0)

            {

                int flag3 = 0;

                for(int j=0 ; j<no\_frames ; j++)

                {

                     temp[j] == -1;

                     for(int k= i+1 ; k<no\_pages ; k++)

                     {

                        if(pages[k] == frames[j])

                        {

                            temp[j] = k;

                            break;

                        }

                     }

                }

                 for(int j=0 ; j<no\_frames ; j++)

            {

                if(temp[j] == -1)

                {

                    pos = j;          // on which frame number page will be inserted

                    flag3= 1;

                    break;

                }

            }

               if(flag3 == 0)

               {

                   int max = temp[0];

                   pos = 0;          // if remove show error

                  for(int j=1 ; j<no\_frames ; j++)

                  {

                     if(temp[j] > max)

                     {

                        max = temp[j];

                        pos = j;

                     }

                  }

               }

            frames[pos] = pages[i];

            faults++;

            }

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

        for(int j=0 ; j<no\_frames ; j++)

        {

            printf("%d \t \t",frames[j]);

        }

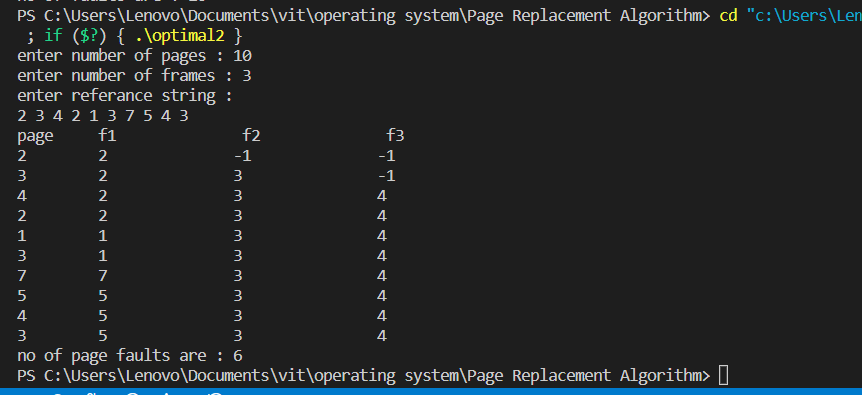
        printf("\n");

        }

      printf("no of page faults are : %d \n",faults);

}

**Output :**

****