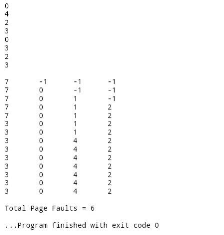
1) First in first out

```
#include<stdio.h>
int main()
{
int i,j,n,a[50],frame[10],no,k,avail,count=0;
            printf("\nenter the length of the Reference string:\n");
   scanf("%d",&n);
            printf("\n enter the reference string:\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("\tref string\t page frames\n");
      for(i=1;i<=n;i++)
                         {
                                    printf("%d\t\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\n");
       }
                        printf("Page Fault Is %d",count);
                        return 0;
}
```

```
enter the length of the Reference string:
7

 enter the reference string:
1
3
0
3
5
6
3

 enter the number of Frames:3
        ref string       page frames
1               1       -1      -1

3               1       3       -1

0               1       3       0

3

5               5       3       0

6               5       6       0

3               5       6       3

Page Fault Is 6

...Program finished with exit code 0
```

2)Optimal page replacement

```
#include<stdio.h>

int main()

{

    int no_of_frames, no_of_pages, frames[10], pages[30], temp[10], flag1, flag2, flag3, i, j, k, pos, max, faults = 0;

    printf("Enter number of frames: ");

    scanf("%d", &no_of_frames);


    printf("Enter number of pages: ");

    scanf("%d", &no_of_pages);


    printf("Enter page reference string: ");


    for(i = 0; i < no_of_pages; ++i){

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

    }


    for(i = 0; i < no_of_frames; ++i){

        frames[i] = -1;

    }


    for(i = 0; i < no_of_pages; ++i){

        flag1 = flag2 = 0;


        for(j = 0; j < no_of_frames; ++j){

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

                   flag1 = flag2 = 1;

                   break;

               }

        }


        if(flag1 == 0){

            for(j = 0; j < no_of_frames; ++j){

                if(frames[j] == -1){

                    faults++;

                    frames[j] = pages[i];

                    flag2 = 1;

                    break;

                }

            }

        }


        if(flag2 == 0){

         flag3 =0;


            for(j = 0; j < no_of_frames; ++j){

             temp[j] = -1;


             for(k = i + 1; k < no_of_pages; ++k){

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

             temp[j] = k;

             break;

             }

             }

            }


            for(j = 0; j < no_of_frames; ++j){

             if(temp[j] == -1){

             pos = j;

             flag3 = 1;

             break;

             }

            }


            if(flag3 ==0){

             max = temp[0];

             pos = 0;


             for(j = 1; j < no_of_frames; ++j){

             if(temp[j] > max){

             max = temp[j];

             pos = j;

             }

             }

            }

frames[pos] = pages[i];

faults++;

        }


        printf("\n");


        for(j = 0; j < no_of_frames; ++j){

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

        }

    }


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


    return 0;

}
```



```
Enter number of frames: 4
Enter number of pages: 14
Enter page reference string: 7
0
1
2
0
3
0
4
2
3
0
3
2
3

7       -1      -1      -1
7       0       -1      -1
7       0       1       -1
7       0       1       2
7       0       1       2
3       0       1       2
3       0       1       2
3       0       4       2
3       0       4       2
3       0       4       2
3       0       4       2
3       0       4       2
3       0       4       2
3       0       4       2

Total Page Faults = 6

...Program finished with exit code 0
```

3) least recently used

```c
#include<stdio.h>

int findLRU(int time[], int n){
int i, minimum = time[0], pos = 0;

for(i = 1; i < n; ++i){
if(time[i] < minimum){
minimum = time[i];
pos = i;
}
}

return pos;
}

int main()
{
int no_of_frames, no_of_pages, frames[10], pages[30], counter = 0, time[10], flag1, flag2, i, j, pos, faults = 0;
printf("Enter number of frames: ");
scanf("%d", &no_of_frames);

printf("Enter number of pages: ");
scanf("%d", &no_of_pages);

printf("Enter reference string: ");

for(i = 0; i < no_of_pages; ++i){
scanf("%d", &pages[i]);
}

for(i = 0; i < no_of_frames; ++i){
frames[i] = -1;
}

for(i = 0; i < no_of_pages; ++i){
flag1 = flag2 = 0;

for(j = 0; j < no_of_frames; ++j){
if(frames[j] == pages[i]){
counter++;
time[j] = counter;
flag1 = flag2 = 1;
break;
}
}

if(flag1 == 0){
for(j = 0; j < no_of_frames; ++j){
if(frames[j] == -1){
counter++;
faults++;
frames[j] = pages[i];
time[j] = counter;
flag2 = 1;
break;
}
}
}

if(flag2 == 0){
pos = findLRU(time, no_of_frames);
counter++;
faults++;
frames[pos] = pages[i];
time[pos] = counter;
}

printf("\n");

for(j = 0; j < no_of_frames; ++j){
printf("%d\t", frames[j]);
}
}

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

return 0;
}
```

```
Enter number of frames: 4
Enter number of pages: 10
Enter reference string: 2
3
4
2
1
3
7
5
4
3

2       -1      -1      -1
2       3       -1      -1
2       3       4       -1
2       3       4       -1
2       3       4       1
2       3       4       1
2       3       7       1
5       3       7       1
5       3       7       4
5       3       7       4

Total Page Faults = 7

...Program finished with exit code 0
```