Ex. No.: 11c)
Date:

Optimal

Aim:
To write a c program to implement Optimal page replacement algorithm.

ALGORITHM:

1. Start the process
2. Declare the size
3. Get the number of pages to be inserted
4. Get the value
5. Declare counter and stack
6. Select the least frequently used page by counter value
7. Stack them according the selection.
8. Display the values
9. Stop the process

PROGRAM:

```
# include <stdio.h>
int main ()
{
    int refstr[100], frames [10];
    int n, f, i, j, k, Page Faults =0, hit;
    Printf ("Enter the size of reference string :")
    scanf("%d", &n);
    for (i=0; i<n; i++){
        Printf ("Enter [%d]: ", i+1);
        scanf ("%d", &refstr);
        for (i=0; i<f; i++)
            frames [i] = -1;
```

```
for (i=0; i<np; i++)
{
  int hit=0;
  for (j=0); j<occ; j++)
  {
    if (frm[j] == pgs[i])
     {
        hit=1;
        break;
     }
  }
  if (!hit)
  {
    fault s++;
    if (occ<nf)
     frm[occ++] = pgs[i];
    else
    int far = i+1; index = -1;
    for (j=0; j<nf; j++)
    {
      int found=0
      for (k=i+1; k<np; k++)
      {
        if (frm[j] == pgs[k])
         {
           if (k > far)
            {
              far = k;
              index = j;
            }
           found = 1;
           break
         }
      }
      if (!found)
      {
        index = j;
        break;
      }
  }
}
```

Output:

Enter no. of. frames : 2
Enter no. of. Pages : 5
Enter page reference string : 1 2 3 2 1

Page	Frames	
1	1	—
2	1	2
3	3	2
2	3	2
1	1	2

Total Page Faults = 4

Result:

Hence the Optimal Page Reference Algorithm is implemented and Verified.

ACE Scanner