

Ex. No.: 11(c)
NAME:SOMILA

Date:15/4/25
ROLLNO:231901052

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 search(int key, int frame[], int f) {    for (int i = 0; i < f; i++) {        if  
(frame[i] == key)            return 1;  
    }    return 0;  
}
```

```
int predict(int pages[], int frame[], int n, int index, int f) {    int res = -1, farthest = index;  
  
    for (int i = 0; i < f; i++) {        int j;
```

```

        for (j = index; j < n; j++) {            if (frame[i] == pages[j]) {
if (j > farthest) {                            farthest = j;                res = i;
    }                                           break;
        }
    }

    // If page not found in future, return that index
    if (j == n)        return i;
}

return (res == -1) ? 0 : res;
}

int main() {
    int n, f;
    printf("Enter number of frames: ");    scanf("%d", &f);

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

    int pages[n];    printf("Enter reference string: ");    for (int i = 0; i < n;
i++)
        scanf("%d", &pages[i]);

    int frame[f];
    int count = 0, index = 0;

    for (int i = 0; i < f; i++)
        frame[i] = -1;

    for (int i = 0; i < n; i++) {        if (search(pages[i], frame, f)) {
        // No page fault        } else {            if (index < f) {
frame[index++] = pages[i];
        } else {
            int pos = predict(pages, frame, n, i + 1, f);                frame[pos] = pages[i];
        }                count++;
    }
}

    for (int j = 0; j < f; j++) {        if (frame[j] != -1)            printf("%d
", frame[j]);
        else                printf("-1 ");
    }
}

```

```
        }    printf("\n");  
    }  
  
    printf("\nTotal Page Faults = %d\n", count);  
    return 0;  
}
```

Output:

Enter number of frames: 3
Enter number of pages: 12
Enter reference string: 7 0 1 2 0 3 0 4 2 3 0 3

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

Total Page Faults = 9

Result:

Thus the algorithm is executed successfully.