

NAME: ARUN MC

ROLL NO: 230701036

Exp:11c

Optimal

Aim:

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

CODE:

```

#include <stdio.h>

int i, j, nof, nor, flag = 0, ref[50], frm[50], pf = 0, victim = -1;
int optcal[50], count = 0;

int optvictim(int index);

int main() {
    printf("\nOPTIMAL PAGE REPLACEMENT ALGORITHM");
    printf("\n-----");

    printf("\nEnter the number of frames: ");
    scanf("%d", &nof);

    printf("Enter the number of reference string elements: ");
    scanf("%d", &nor);

    printf("Enter the reference string: ");
    for (i = 0; i < nor; i++)
        scanf("%d", &ref[i]);

    // Initialize frame and calculation arrays
    for (i = 0; i < nof; i++) {
        frm[i] = -1;
        optcal[i] = 0;
    }

    printf("\nReference String:\n");
    for (i = 0; i < nor; i++)
        printf("%4d", ref[i]);

    printf("\n\nProcessing...\n");

    for (i = 0; i < nor; i++) {
        flag = 0;
        printf("\nref no %2d ->\t", ref[i]);

        // Check if page already in frame
        for (j = 0; j < nof; j++) {
            if (frm[j] == ref[i]) {
                flag = 1;
                break;
            }
        }

        if (flag == 0) {
            count++;
            if (count <= nof)

```

"optimal.c" 98L, 2193C

```

        victim = optvictim(i); // Find optimal victim

        frm[victim] = ref[i];
        pf++; // Page fault
    }

    // Display current frame state
    for (j = 0; j < nof; j++) {
        if (frm[j] != -1)
            printf("%4d", frm[j]);
        else
            printf("   -");
    }
}

printf("\n\nTotal Page Faults: %d\n", pf);

return 0;
}

int optvictim(int index) {
    int i, j, temp, notfound;

    for (i = 0; i < nof; i++) {
        notfound = 1;
        for (j = index + 1; j < nor; j++) {
            if (frm[i] == ref[j]) {
                notfound = 0;
                optcal[i] = j;
                break;
            }
        }

        if (notfound == 1)
            return i;
    }

    // Find frame with farthest next use
    temp = optcal[0];
    int pos = 0;
    for (i = 1; i < nof; i++) {
        if (optcal[i] > temp) {
            temp = optcal[i];
            pos = i;
        }
    }

    return pos;
}

```

OUTPUT:

```
OPTIMAL PAGE REPLACEMENT ALGORITHM
-----
Enter the number of frames: 6
Enter the number of reference string elements: 6 5 4 3 2 1
Enter the reference string: 3

Reference String:
  5   4   3   2   1   3

Processing...

ref no  5 ->      5   -   -   -   -   -
ref no  4 ->      5   4   -   -   -   -
ref no  3 ->      5   4   3   -   -   -
ref no  2 ->      5   4   3   2   -   -
ref no  1 ->      5   4   3   2   1   -
ref no  3 ->      5   4   3   2   1   -

Total Page Faults: 5
```