

Ex. No.: 11c)

Date: 18/4/25

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 to the selection.
8. Display the values
9. Stop the process

PROGRAM:

```
#include <stdio.h>
int main()
{
    int frames, pages, i, j, k, pages - faults = 0, ref string [100];
    int memory [10], future [10], flags, flag 2, pos;
    Print ("Enter no of frames: ");
    scanf ("%d", &frames);
    Print ("Enter number of pages: ");
    scanf ("%d", &pages);
    Print ("Enter the reference string: ");
    for (i = 0; i < pages; i++) {
        scanf ("%d", &ref-str[i]);
    }
}
```



```
Print("\n");
```

```
for("i=0; i < pages; i++) {
```

```
flag1 = flag2 = 0;
```

```
for (j=0; j < frames; j++) {
```

```
if (memory[i] == ref-str[i]) {
```

```
flag1 = flag2 = 1;
```

```
break;
```

```
}
```

```
if (flag1 == 0) {
```

```
for (j=0; j < frames; j++) {
```

```
(if (memory[j] == -1) {
```

```
memory[j] = ref-str[i];
```

```
page-faults++;
```

```
flag2 = 1;
```

```
break;
```

```
}
```

```
if (flag2 == 0) {
```

```
for (j=0; j < frames; j++) {
```

```
future[j] = -1;
```

```
for (k=i+1; k < pages; k++) {
```

```
if (memory[j] == ref-str[k]) {
```

```
future[j] = k;
```

```
break;
```

```
}
```

```
}
```

```
}
```


Enter number of frames: 2

Enter number of pages: 6

Enter reference string: 1 5 6 4 5 5

1 -
1 5
6 5
4 5
4 5
4 5

Total page faults = 4

```
int { max = -1;  
for (j=0; j < frames; j++) {  
    if (future [j] == -1) {
```

Output:

```
        pos = j;  
    }  
    if (future [j] > max) {  
        max = future [j];  
        pos = j;  
    }
```

```
    memory [pos] = ref_str [i];  
    page_faults++;  
}
```

```
for (j=0; j < frames; j++) {  
    if (memory [j] == -1)  
        print (" ");
```

```
    else  
        print (" ", memory [j]);
```

```
    print ("\n");  
}
```

```
print ("Total page faults = ", page_faults);  
return 0;
```

```
}
```

Result:

Hence, a C program to implement optimal page replacement algorithm has been executed successfully.