Code 14

```
#include <stdio.h>

#define MAX_FRAMES 10
#define MAX_PAGES 100

int predict(int pages[], int frames[], 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 (frames[i] == pages[j]) {
                if (j > farthest) {
                    farthest = j;
                    res = i;
                }
                break;
            }
        }
        // If the page is never used again
        if (j == n)
            return i;
    }

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

```
int main() {
    int pages[MAX_PAGES], frames[MAX_FRAMES];
    int n, f, faults = 0;
    int i, j, hit;

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

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

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

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

    for(i = 0; i < n; i++) {
        hit = 0;

        // Check if page is already in frame
        for(j = 0; j < f; j++) {
            if(frames[j] == pages[i]) {
                hit = 1;
                break;
```

```
    }
}

if(!hit) {
    int empty = -1;
    for(j = 0; j < f; j++) {
        if(frames[j] == -1) {
            empty = j;
            break;
        }
    }

    if(empty != -1) {
        frames[empty] = pages[i];
    } else {
        int pos = predict(pages, frames, n, i + 1, f);
        frames[pos] = pages[i];
    }

    faults++;
}

printf("Frames after accessing page %d: ", pages[i]);
for(j = 0; j < f; j++) {
    if(frames[j] != -1)
        printf("%d ", frames[j]);
    else
        printf("- ");
```

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

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

    return 0;
}
```