```c
#include <limits.h>
#include <stdio.h>

#define MAX_FRAMES 3
#define MAX_PAGES 10

int frames[MAX_FRAMES], pages[MAX_PAGES];
int num_pages;

void initialize_frames() {
  for (int i = 0; i < MAX_FRAMES; i++) {
    frames[i] = -1;
  }
}

void print_frames() {
  for (int i = 0; i < MAX_FRAMES; i++) {
    if (frames[i] == -1) {
      printf("- ");
    } else {
      printf("%d ", frames[i]);
    }
  }
  printf("\n");
}

int find_optimal_index(int current_index) {
  int furthest_use = -1;
  int optimal_index = 0;

  for (int i = 0; i < MAX_FRAMES; i++) {
    int j;
    for (j = current_index; j < num_pages; j++) {
      if (frames[i] == pages[j]) {
        if (j > furthest_use) {
          furthest_use = j;
          optimal_index = i;
        }
        break;
      }
    }
    if (j == num_pages) {
      return i;
    }
  }
  return optimal_index;
}

void optimal_page_replacement() {
  int current_index = 0;

  for (int i = 0; i < num_pages; i++) {
    int page = pages[i];
    int found = 0;

    for (int j = 0; j < MAX_FRAMES; j++) {
      if (frames[j] == page) {
        found = 1;
        break;
      }
    }

    if (!found) {
```

```
      int optimal_index = find_optimal_index(i + 1);
      frames[optimal_index] = page;
    }

    print_frames();
  }
}

int main() {
  printf("Enter number of pages: ");
  scanf("%d", &num_pages);

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

  initialize_frames();
  optimal_page_replacement();

  return 0;
}
```