```
#include <stdio.h>
#define MAX 3

typedef struct queue {
   int aiframe[MAX];
   int last_used[MAX];
} queue;

void fnLRU(int[], int);

int main() {
   int size;
   printf("******* LRU Page Replacement Algorithm *******\n");
   printf("Enter the size: ");
   scanf("%d", &size);
   int arr[size];
   printf("Enter the elements: ");
   for (int i = 0; i < size; i++) {
      scanf("%d", &arr[i]);
   }
   fnLRU(arr, size);
   return 0;
}

void fnLRU(int arr[], int size) {
   int pg = 0;
   queue q;

   for (int i = 0; i < MAX; i++) {
      q.aiframe[i] = -1;
      q.last_used[i] = -1;
```

```
}

int time = 0;
for (int i = 0; i < size; i++) {
   int flag = 0;



   for (int j = 0; j < MAX; j++) {
      if (q.aiframe[j] == arr[i]) {
         flag = 1;
         q.last_used[j] = time++;
         break;
      }
   }



   if (flag == 0) {
      int lru_index = 0;
      for (int j = 1; j < MAX; j++) {
         if (q.last_used[j] < q.last_used[lru_index]) {
            lru_index = j;
         }
      }



      q.aiframe[lru_index] = arr[i];
      q.last_used[lru_index] = time++;
      pg++;
   }

   printf("Frame: ");
```

```
        for (int j = 0; j < MAX; j++) {
            if (q.aiframe[j] != -1) {
                printf("%d ", q.aiframe[j]);
            } else {
                printf("- ");
            }
        }
        printf("\n");
    }

    printf("Total Number of Page faults (LRU): %d\n", pg);
    printf("Page Fault Frequency (LRU): %.2f%%\n", (float)pg / size * 100);
    printf("Hit Frequency (LRU): %.2f%%\n", 100 - (float)pg / size * 100);
}
```