Program:

```
#include <iostream>
#include <vector>
using namespace std;
int LRU(int pages[], int n, int frames) {
    vector<int> memory; // Represents frames in memory
    int pageFaults = 0; // Counter for page faults

    for (int i = 0; i < n; i++) {
        // Check if the page is already in memory
        bool found = false;
        for (int j = 0; j < memory.size(); j++) {
            if (memory[j] == pages[i]) {
                found = true;
                break;
            }
        }
 // If page is not in memory, handle page fault
        if (!found) {
            pageFaults++; // Increment page fault counter
  if (memory.size() == frames) {
                // Find the least recently used page
                int lruIndex = 0;
                int farthest = i - 1; // Track the farthest usage

                for (int j = 0; j < memory.size(); j++) {
                    int k;
                    for (k = i - 1; k >= 0; k--) {
                        if (memory[j] == pages[k]) {
                            if (k < farthest) {
                                farthest = k;
                                lruIndex = j;
                            }
                            break;
                        }
                    }
                    // If a page is not used in the past, it is the LRU page
                    if (k == -1) {
                        lruIndex = j;
                        break;
                    }
                }

                // Replace the LRU page with the new page
                memory[lruIndex] = pages[i];
            } else {
                // If there is space in memory, simply add the page
                memory.push_back(pages[i]);
            }
        }
    }

    return pageFaults;
}

int main() {
    int n, frames;
    cout << "Enter the number of pages: ";
    cin >> n;
    int pages[n];
    cout << "Enter the page reference string: ";
    for (int i = 0; i < n; i++) {
        cin >> pages[i];
    }

    cout << "Enter the number of frames: ";
    cin >> frames;

    int pageFaults = LRU(pages, n, frames);
    cout << "Total Page Faults: " << pageFaults << endl;

    return 0;
}
```

Output:

```
Enter the number of pages: 12
Enter the page reference string: 1 2 3 4 1 2 5 1 2 3 4 5
Enter the number of frames: 3
Total Page Faults: 10
```

Conclusion: Least Recently Used Page Replacement Algorithm program was implemented successfully.