LAB 9

NAME: RIA JAYANT TAMBVE
ROLL NO: 57

Aim: Implement the all page replacement algorithms
a) FIFO b) LRU c) Optimal

Code:

```cpp
#include <iostream>
#include <algorithm>
using namespace std;

bool pageExists(int frames[], int n, int page) {
    for (int i = 0; i < n; i++) {
        if (frames[i] == page) return true;
    }
    return false;
}

void fifoPageReplacement(int pages[], int n, int capacity) {
    int frames[capacity] = {-1};
    int front = 0, pageFaults = 0;

    cout << "FIFO Page Replacement\n";
    for (int i = 0; i < n; i++) {
        if (!pageExists(frames, capacity, pages[i])) {
            frames[front] = pages[i];
            front = (front + 1) % capacity;
            pageFaults++;
        }
        cout << "Frames after accessing page " << pages[i] << ": ";
        for (int j = 0; j < capacity; j++) {
            cout << (frames[j] == -1 ? "_" : to_string(frames[j])) << " ";
        }
        cout << endl;
    }
    cout << "Total Page Faults: " << pageFaults << endl << endl;
}

void lruPageReplacement(int pages[], int n, int capacity) {
    int frames[capacity] = {-1};
    int recent[capacity] = {0}; // Tracks recent access times
    int time = 0, pageFaults = 0;

    cout << "LRU Page Replacement\n";
    for (int i = 0; i < n; i++) {
        time++;
        if (!pageExists(frames, capacity, pages[i])) {
```

```
            int lruIndex = 0, minTime = time;
            for (int j = 0; j < capacity; j++) {
                if (frames[j] == -1) {
                    lruIndex = j;
                    break;
                }
                if (recent[j] < minTime) {
                    minTime = recent[j];
                    lruIndex = j;
                }
            }
            frames[lruIndex] = pages[i];
            pageFaults++;
        }
        for (int j = 0; j < capacity; j++) {
            if (frames[j] == pages[i]) {
                recent[j] = time;
                break;
            }
        }
        cout << "Frames after accessing page " << pages[i] << ": ";
        for (int j = 0; j < capacity; j++) {
            cout << (frames[j] == -1 ? "_" : to_string(frames[j])) << " ";
        }
        cout << endl;
    }
    cout << "Total Page Faults: " << pageFaults << endl << endl;
}

void optimalPageReplacement(int pages[], int n, int capacity) {
    int frames[capacity] = {-1};
    int pageFaults = 0;

    cout << "Optimal Page Replacement\n";
    for (int i = 0; i < n; i++) {
        if (!pageExists(frames, capacity, pages[i])) {
            int replaceIndex = -1, farthest = i + 1;
            for (int j = 0; j < capacity; j++) {
                if (frames[j] == -1) {
                    replaceIndex = j;
```

```
                    break;
                }
                int nextUse = n; // Assume page won't be used again
                for (int k = i + 1; k < n; k++) {
                    if (frames[j] == pages[k]) {
                        nextUse = k;
                        break;
                    }
                }
                if (nextUse > farthest) {
                    farthest = nextUse;
                    replaceIndex = j;
                }
            }
            frames[replaceIndex] = pages[i];
            pageFaults++;
        }
        cout << "Frames after accessing page " << pages[i] << ": ";
        for (int j = 0; j < capacity; j++) {
            cout << (frames[j] == -1 ? "_" : to_string(frames[j])) << " ";
        }
        cout << endl;
    }
    cout << "Total Page Faults: " << pageFaults << endl << endl;
}

int main() {
    int n;
    cout << "Enter the number of pages: ";
    cin >> n;

    int pages[n];
    cout << "Enter the page sequence: ";
    for (int i = 0; i < n; i++) {
        cin >> pages[i];
    }

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

    fifoPageReplacement(pages, n, capacity);
    lruPageReplacement(pages, n, capacity);
    optimalPageReplacement(pages, n, capacity);

    return 0;
}
```

Output:

```
PS C:\Users\HP\Desktop\pract>  & 'c:\Users\HP\.vscode\extensions\ms-vscode.
debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-
crosoft-MIEngine-Out-ipc0h2gk.lo3' '--stderr=Microsoft-MIEngine-Error-kn5b3
MIEngine-Pid-5z4bfz5b.jjf' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--inter
Enter the number of pages: 13
Enter the page sequence: 7 0 1 2 0 3 0 4 2 3 0 3 2
Enter the number of frames: 3
FIFO Page Replacement
Frames after accessing page 7: 7 0 0
Frames after accessing page 0: 7 0 0
Frames after accessing page 1: 7 1 0
Frames after accessing page 2: 7 1 2
Frames after accessing page 0: 0 1 2
Frames after accessing page 3: 0 3 2
Frames after accessing page 0: 0 3 2
Frames after accessing page 4: 0 3 4
Frames after accessing page 2: 2 3 4
Frames after accessing page 3: 2 3 4
Frames after accessing page 0: 2 0 4
Frames after accessing page 3: 2 0 3
Frames after accessing page 2: 2 0 3
Total Page Faults: 9
LRU Page Replacement
Frames after accessing page 7: 7 0 0
Frames after accessing page 0: 7 0 0
Frames after accessing page 1: 7 0 1
Frames after accessing page 2: 2 0 1
Frames after accessing page 0: 2 0 1
Frames after accessing page 3: 2 0 3
Frames after accessing page 0: 2 0 3
Frames after accessing page 4: 4 0 3
Frames after accessing page 2: 4 0 2
Frames after accessing page 3: 4 3 2
Frames after accessing page 0: 0 3 2
Frames after accessing page 3: 0 3 2
Frames after accessing page 2: 0 3 2
Total Page Faults: 8
Optimal Page Replacement
Frames after accessing page 7: 7 0 0
Frames after accessing page 0: 7 0 0
Frames after accessing page 1: 1 0 0
Frames after accessing page 2: 2 0 0
Frames after accessing page 0: 2 0 0
Frames after accessing page 3: 3 0 0
Frames after accessing page 0: 3 0 0
Frames after accessing page 4: 3 4 0
Frames after accessing page 2: 3 2 0
Frames after accessing page 3: 3 2 0
Frames after accessing page 0: 3 2 0
Frames after accessing page 3: 3 2 0
Frames after accessing page 2: 3 2 0
Total Page Faults: 6
```