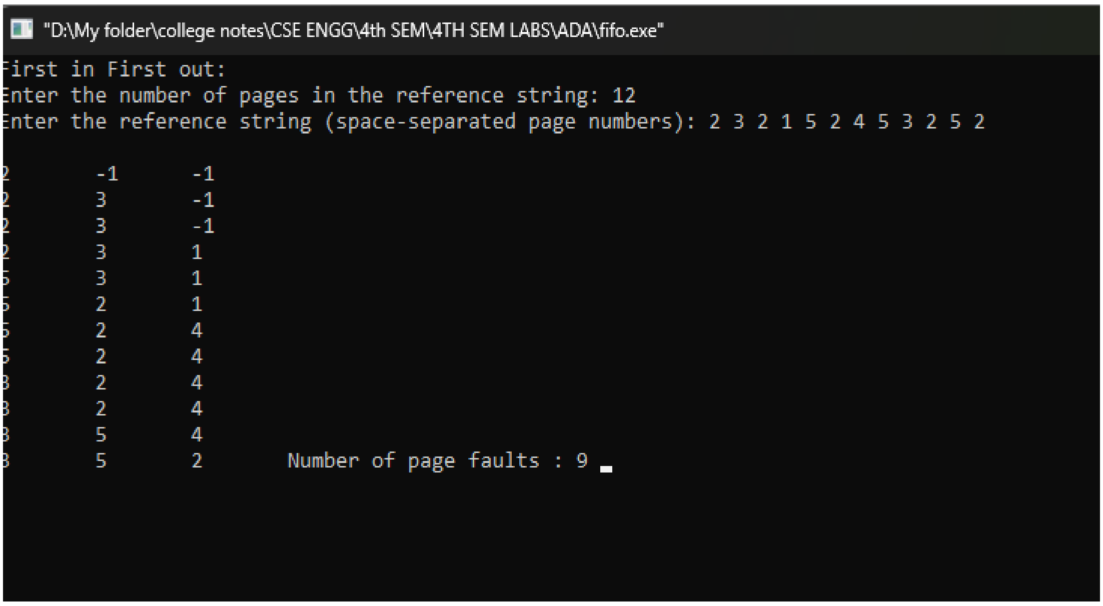
LAB 9

Q. Write a C program to simulate page replacement algorithms.

1. FIFO

```
#include <stdio.h>
#include <conio.h>

int fr[3];

void display() {
    int i;
    printf("\n");
    for (i = 0; i < 3; i++)
        printf("%d\t", fr[i]);
}

int main() {
    void display();
    int i, j, page[12], n;
    int flag1 = 0, flag2 = 0, pf = 0, frsize = 3, top = 0;
    printf("First in First out:\n");
    printf("Enter the number of pages in the reference string: ");
    scanf("%d", &n);

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

    for (i = 0; i < 3; i++) {
        fr[i] = -1;
    }

    for (j = 0; j < n; j++) {
        flag1 = 0;
        flag2 = 0;
        for (i = 0; i < 3; i++) {
            if (fr[i] == page[j]) {
                flag1 = 1;
                flag2 = 1;
                break;
            }
        }

        if (flag1 == 0) {
            for (i = 0; i < frsize; i++) {
                if (fr[i] == -1) {
                    fr[i] = page[j];
                    flag2 = 1;
                    break;
                }
            }
        }

        if (flag2 == 0) {
            fr[top] = page[j];
            top = (top + 1) % frsize;
            pf++;
        }

        display();
    }

    printf("Number of page faults : %d ", pf + frsize);
    getch();

    return 0;
}
```

OUTPUT:



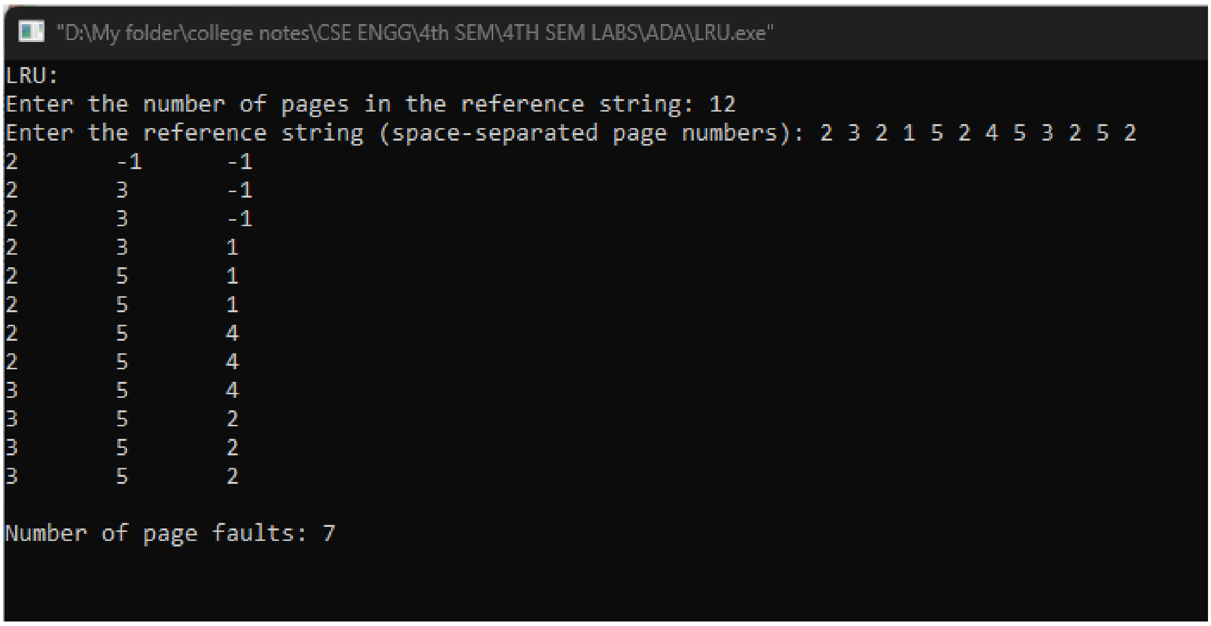
"D:\My folder\college notes\CSE ENGG\4th SEM\4TH SEM LABS\ADA\fifo.exe"

```
First in First out:
Enter the number of pages in the reference string: 12
Enter the reference string (space-separated page numbers): 2 3 2 1 5 2 4 5 3 2 5 2

2       -1      -1
2       3       -1
2       3       -1
2       3       1
5       3       1
5       2       1
5       2       4
5       2       4
3       2       4
3       2       4
3       5       4
3       5       2       Number of page faults : 9
```

2. LRU

```
#include <stdio.h>
#include <conio.h>

void display(int fr[], int frsize) {
    for (int i = 0; i < frsize; i++) {
        if (fr[i] == -1) {
            printf("-1\t");
        } else {
            printf("%d\t", fr[i]);
        }
        if ((i + 1) % 3 == 0) {
            printf("\n");
        }
    }
}

int main() {
    int fr[3];
    int page[12], n;
    int fs[3];
    int index, k, l, flag1, flag2, pf, frsize = 3;
    printf("LRU:\n");
    printf("Enter the number of pages in the reference string: ");
    scanf("%d", &n);

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

    for (int i = 0; i < 3; i++) {
        fr[i] = -1;
    }

    flag1 = 0;
    flag2 = 0;
    pf = 0;

    for (int j = 0; j < n; j++) {
        flag1 = 0;
        flag2 = 0;
        for (int i = 0; i < 3; i++) {
            if (fr[i] == page[j]) {
                flag1 = 1;
                flag2 = 1;
                break;
            }
        }
        if (flag1 == 0) {
            for (int i = 0; i < frsize; i++) {
                if (fr[i] == -1) {
                    fr[i] = page[j];
                    flag2 = 1;
                    break;
                }
            }
        }

        if (flag2 == 0) {
            for (int i = 0; i < 3; i++) {
                fs[i] = 0;
            }

            for (int k = j - 1, l = 1; l <= frsize - 1; l++, k--) {
                for (int i = 0; i < 3; i++) {
                    if (fr[i] == page[k]) {
                        fs[i] = 1;
                    }
                }
            }

            index = -1;
            for (int i = 0; i < 3; i++) {
                if (fs[i] == 0) {
                    index = i;
                    break;
                }
            }

            if (index == -1) {
                index = 0;
            }

            fr[index] = page[j];
            pf++;
        }

        display(fr, frsize);
    }

    printf("\nNumber of page faults: %d\n", pf + frsize);
    getch();

    return 0;
}
```

OUTPUT:



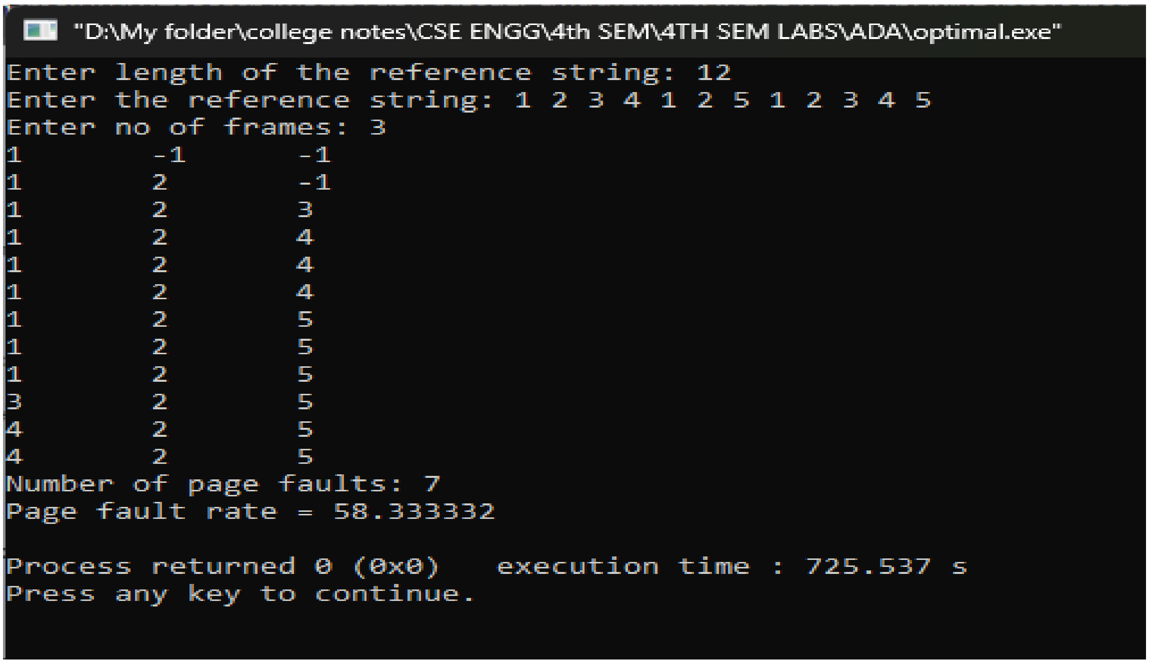
"D:\My folder\college notes\CSE ENGG\4th SEM\4TH SEM LABS\ADA\LRU.exe"

```
LRU:
Enter the number of pages in the reference string: 12
Enter the reference string (space-separated page numbers): 2 3 2 1 5 2 4 5 3 2 5 2
2       -1      -1
2       3       -1
2       3       -1
2       3       1
2       5       1
2       5       1
2       5       4
2       5       4
3       5       4
3       5       2
3       5       2
3       5       2

Number of page faults: 7
```

3. OPTIMAL

```
#include <stdio.h>

int fr[3], n, m;

void display() {
    for (int i = 0; i < m; i++)
        printf("%d\t", fr[i]);
    printf("\n");
}

int main() {
    int i, j, page[20], fs[10];
    int max, found = 0, lg[3], index, k, flag1 = 0, flag2 = 0, pf = 0;
    float pr;

    printf("Enter length of the reference string: ");
    scanf("%d", &n);

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

    printf("Enter no of frames: ");
    scanf("%d", &m);

    for (i = 0; i < m; i++)
        fr[i] = -1;

    pf = m;

    for (j = 0; j < n; j++) {
        flag1 = 0;
        flag2 = 0;

        for (i = 0; i < m; i++) {
            if (fr[i] == page[j]) {
                flag1 = 1;
                flag2 = 1;
                break;
            }
        }

        if (flag1 == 0) {
            for (i = 0; i < m; i++) {
                if (fr[i] == -1) {
                    fr[i] = page[j];
                    flag2 = 1;
                    break;
                }
            }
        }

        if (flag2 == 0) {
            for (i = 0; i < m; i++)
                lg[i] = 0;

            for (i = 0; i < m; i++) {
                for (k = j + 1; k < n; k++) {
                    if (fr[i] == page[k]) {
                        lg[i] = k - j;
                        break;
                    }
                }
            }

            found = 0;
            for (i = 0; i < m; i++) {
                if (lg[i] == 0) {
                    index = i;
                    found = 1;
                    break;
                }
            }

            if (found == 0) {
                max = lg[0];
                index = 0;
                for (i = 0; i < m; i++) {
                    if (max < lg[i]) {
                        max = lg[i];
                        index = i;
                    }
                }
            }

            fr[index] = page[j];
            pf++;
        }

        display();
    }

    printf("Number of page faults: %d\n", pf);
    pr = (float)pf / n * 100;
    printf("Page fault rate = %f\n", pr);

    return 0;
}
```

OUTPUT:



"D:\My folder\college notes\CSE ENGG\4th SEM\4TH SEM LABS\ADA\optimal.exe"

```
Enter length of the reference string: 12
Enter the reference string: 1 2 3 4 1 2 5 1 2 3 4 5
Enter no of frames: 3
1       -1      -1
1       2       -1
1       2       3
1       2       4
1       2       4
1       2       4
1       2       5
1       2       5
1       2       5
3       2       5
4       2       5
4       2       5
Number of page faults: 7
Page fault rate = 58.333332

Process returned 0 (0x0)   execution time : 725.537 s
Press any key to continue.
```