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

bool check(int *present, int noFrames, int e){
    for(int i=0; i<noFrames; i++){
        if(present[i]==e) return true;
    }
    return false;
}

void FIFOPageRepAlgo(int *pages, int noPages, int noFrames){
    int chance=0, miss=0, hits=0;

    int *present = new int[noFrames];
    for(int i=0; i<noFrames; i++) present[i]=-1;

    // declare a chart for printing
    int **chart = new int*[noFrames+2];
    for(int i=0; i<noFrames+2; i++){
        chart[i] = new int[noPages];
        for(int j=0; j<noPages; j++){
            chart[i][j]=-1;
        }
    }

    for(int i=0; i<noPages; i++){
        chart[0][i] = pages[i];
    }

    int k=0;
    for(int i=0; i<noPages; i++){

        bool missOrHit=true;

        // if page no was not found in any of the frames
        // miss case
        if(!check(present, noFrames, pages[i])){
            present[chance]=pages[i];
            chance=(chance+1)%noFrames;
            missOrHit=false;
            miss++;
        }
        // hit case
        else{
```

```
            hits++;
        }

        // add the values in the chart
        int j;
        for(j=0; j<noFrames; j++){
            chart[j+1][k] = present[j];
        }

        // update miss or hit in chart
        missOrHit ? chart[j+1][k]=1 : chart[j+1][k]=0;
        k++;
    }

    cout<<endl<<endl<<"Page Fault Details : "<<endl<<endl;

    // Printing the chart
    int NOH = (7*noPages)+1;

    // First row
    for(int j=0; j<noPages; j++){
        printf("   %2d  ", chart[0][j]);
    }
    cout<<endl;
    for(int k=0; k<NOH; k++){
        cout<<"-";
    }
    cout<<endl;

    // middle portion
    for(int i=1; i<noFrames+1; i++){
        for(int j=0; j<noPages; j++){
            if(chart[i][j]==-1) printf("|      ", chart[i][j]);
            else printf("|  %2d  ", chart[i][j]);
        }
        cout<<"|"<<endl;
        for(int k=0; k<NOH; k++){
            cout<<"-";
        }
        cout<<endl;
    }


    // last row
    for(int j=0; j<noPages; j++){
        if(chart[noFrames+1][j]==1) cout<<"| hit  ";
```

```
        else cout<<"| miss ";
    }
    cout<<"|"<<endl;
    for(int k=0; k<NOH; k++){
        cout<<"-";
    }

    cout<<endl<<endl<<"Average Page Fault : "<<((float)miss/noPages)<<" or
"<<miss<<"/"<<noPages<<endl<<endl;
}

int main(){
    int noPages, noFrames;

    cout<<"\n\nName : Mohd Adil \nRoll No : 20BCS042";
    cout<<"\n\nEnter No of Pages and Frames : ";
    cin>>noPages>>noFrames;

    int *pages = new int[noPages];
    cout<<"\nEnter the Pages : ";
    for(int i=0; i<noPages; i++) cin>>pages[i];

    FIFOPageRepAlgo(pages, noPages, noFrames);
    return 0;
}

// sample input:
// 14 4 7 0 1 2 0 3 0 4 2 3 0 3 2 3
```

**OUTPUT**

```
Name : Mohd Adil
Roll No : 20BCS042

Enter No of Pages and Frames : 14 4

Enter the Pages : 7 0 1 2 0 3 0 4 2 3 0 3 2 3


Page Fault Details :
   7     0     1     2     0     3     0     4     2     3     0     3     2     3
-------------------------------------------------------------------------------------
|  7  |  7  |  7  |  7  |  7  |  3  |  3  |  3  |  3  |  3  |  3  |  3  |  3  |  3  |
-------------------------------------------------------------------------------------
|     |  0  |  0  |  0  |  0  |  0  |  0  |  4  |  4  |  4  |  4  |  4  |  4  |  4  |
-------------------------------------------------------------------------------------
|     |     |  1  |  1  |  1  |  1  |  1  |  1  |  1  |  1  |  0  |  0  |  0  |  0  |
-------------------------------------------------------------------------------------
|     |     |     |  2  |  2  |  2  |  2  |  2  |  2  |  2  |  2  |  2  |  2  |  2  |
-------------------------------------------------------------------------------------
| miss | miss | miss | miss | hit  | miss | hit  | miss | hit  | hit  | miss | hit  | hit  | hit  |
-------------------------------------------------------------------------------------

Average Page Fault : 0.5 or 7/14
```