U18CO018
Shubham Shekhaliya
Assignment – 7 (OS)

1-> To implement Shortest Seek Time First Disk Scheduling Algorithm.

Code:-

```
#include<bits/stdc++.h>
using namespace std;
int main() {
    int n;
    cout << "Enter number of Requests \n";
    cin >> n;
    cout << "Enter a head \n";
    int head;
    cin >> head;
    list<int> req;
    cout << "Enter All Requests one by one \n";
    for (int i = 0; i < n; i++) {
        int x;
        cin >> x;
        req.push_back(x);
    }
    vector<int> sequence;
    sequence.push_back(head);
    int seekCount = 0;
    while (req.size() > 0) {
        auto index = req.begin();
        int minimum = INT_MAX;
        for (auto it = req.begin(); it != req.end(); it++) {
            if (abs(head - *it) < minimum) {
                index = it;
                minimum = abs(head - *it);
            }
        }
        seekCount += minimum;
        head = *index;
        sequence.push_back(*index);
        req.erase(index);
    }
```

```
    cout << "Total number of seek operations = " << seekCount << endl;

    cout << "Seek Sequence is \n";

    for (auto it:sequence) {
        cout << it << endl;
    }
    return 0;
}
```

Output:-

```
D:\6th SEM\OS>cd "d:\6th SEM\OS\Assignment7\" && g++ First.cpp -o First && "d:\6th SEM\OS\Assignment7\"First
Enter number of Requests
10
Enter a head
80
Enter All Requests one by one
115 90 85 41 12 35 145 120 156 56
Total number of seek operations = 220
Seek Sequence is
80
85
90
115
120
145
156
56
41
35
12
```

2-> To implement SCAN algorithm for Disk Scheduling.

Code:-

```
#include<bits/stdc++.h>
using namespace std;
const int maxDisk = 200;
int main() {
    int n;
    cout << "Enter number of Requests \n";
    cin >> n;

    int head;
    cout << "Enter a head \n";
    cin >> head;

    int direction;
```

```
    cout << "Enter Direction 0 for toward left 1 for toward right \n";
    cin >> direction;

    vector<int> left, right;
    if (direction == 0) left.push_back(0);
    else right.push_back(maxDisk - 1);

    cout << "Enter All Requests one by one \n";
    for (int i = 0; i < n; i++) {
        int x;
        cin >> x;
        if (x < head) left.push_back(x);
        else if (x > head) right.push_back(x);
    }

    sort(left.begin(), left.end());
    sort(right.begin(), right.end());

    vector<int> sequence;
    sequence.push_back(head);
    int seekCount = 0;
    for (int i = 0; i < 2; i++) {
        if (direction == 0) {  // left
            for (int i = left.size() - 1; i >= 0; i--) {
                sequence.push_back(left[i]);
                seekCount += abs(left[i] - head);
                head = left[i];
            }
        } else {   // right
            for (int i = 0; i < right.size(); i++) {
                sequence.push_back(right[i]);
                seekCount += abs(right[i] - head);
                head = right[i];
            }
        }
        direction ^= 1;
    }
    cout << "Total number of seek operations = " << seekCount << endl;

    cout << "Seek Sequence is \n";

    for (auto it:sequence)
        cout << it << endl;
    return 0;
}
```

Output:-

```
d:\6th SEM\OS\Assignment7>cd "d:\6th SEM\OS\Assignment7\" && g++ Second.cpp -o Second && "d:\6th SEM\OS\Assignment7\"Second
Enter number of Requests
10
Enter a head
80
Enter Direction 0 for toward left 1 for toward right
1
Enter All Requests one by one
115 90 85 41 12 35 145 120 156 56
Total number of seek operations = 306
Seek Sequence is
80
85
90
115
120
145
156
199
56
41
35
12
```