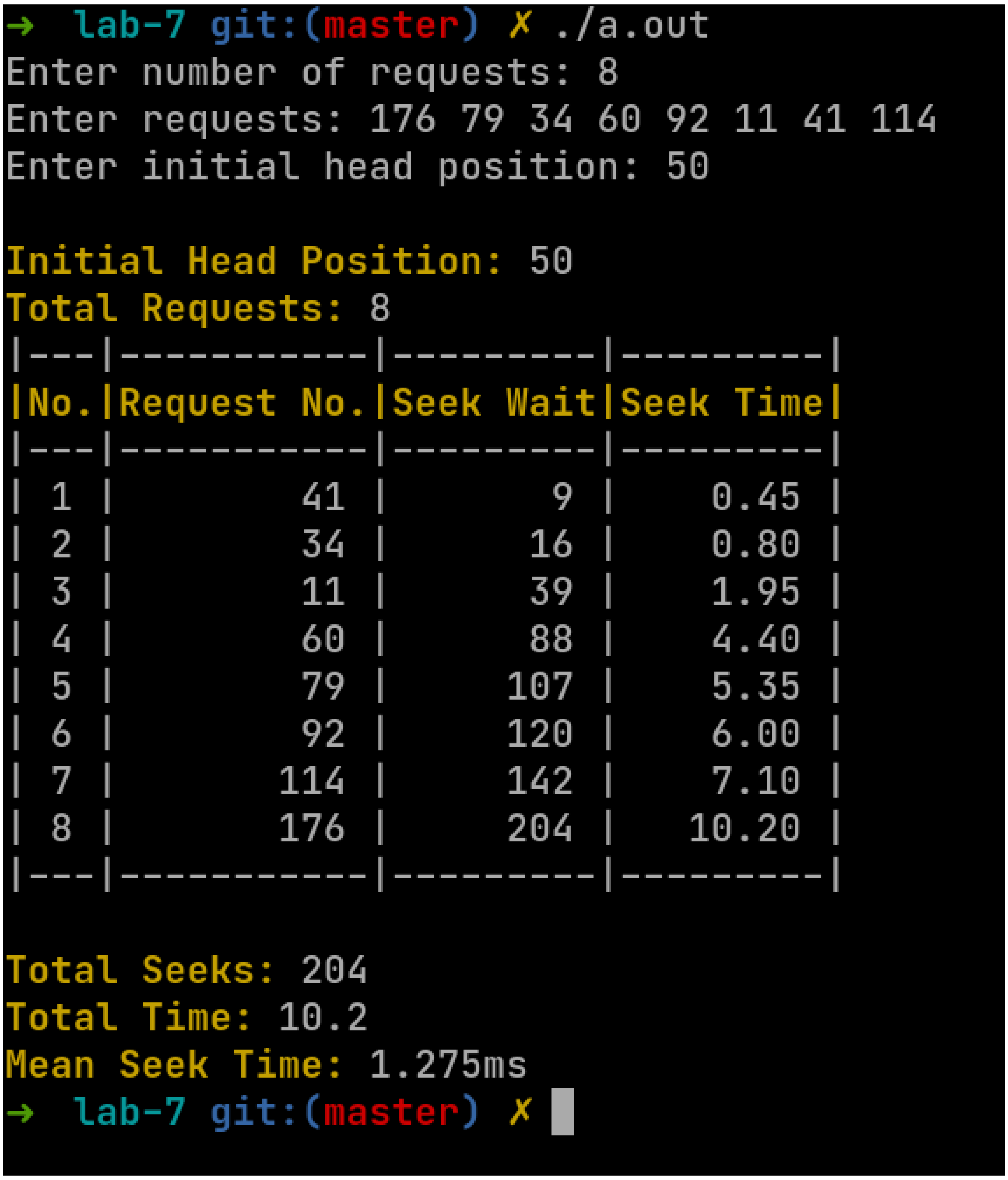
OS LAB 7

SAHIL BONDRE: U18CO021

1. To implement Shortest Seek Time First (SSTF) Disk Scheduling Algorithm

```
#include <stdio.h>

#include <iostream>
#include <set>
#include <vector>

using namespace std;

const int DISK_SIZE = 256;
const double SEEK_DELAY = 0.05;
const string BLUE_PREFIX = "\033[1;33m";
const string BLUE_POSTFIX = "\033[0m";

multiset<int>::iterator closest_to_head(const multiset<int> &requests,
                                        int head) {
  auto closest = requests.begin();
  int closest_distance = abs(head - *closest);

  for (auto it = requests.begin(); it != requests.end(); ++it) {
    if (abs(head - *it) < closest_distance) {
      closest_distance = abs(head - *it);
      closest = it;
    }
  }

  return closest;
}

void sstf(multiset<int> &requests, int start) {
  int head = start;
  int total_seeks = 0;
  int n = requests.size();
  cout << BLUE_PREFIX << "\nInitial Head Position: " << BLUE_POSTFIX <<
head
       << "\n";

  cout << BLUE_PREFIX << "Total Requests: " << BLUE_POSTFIX << n <<
"\n";
```

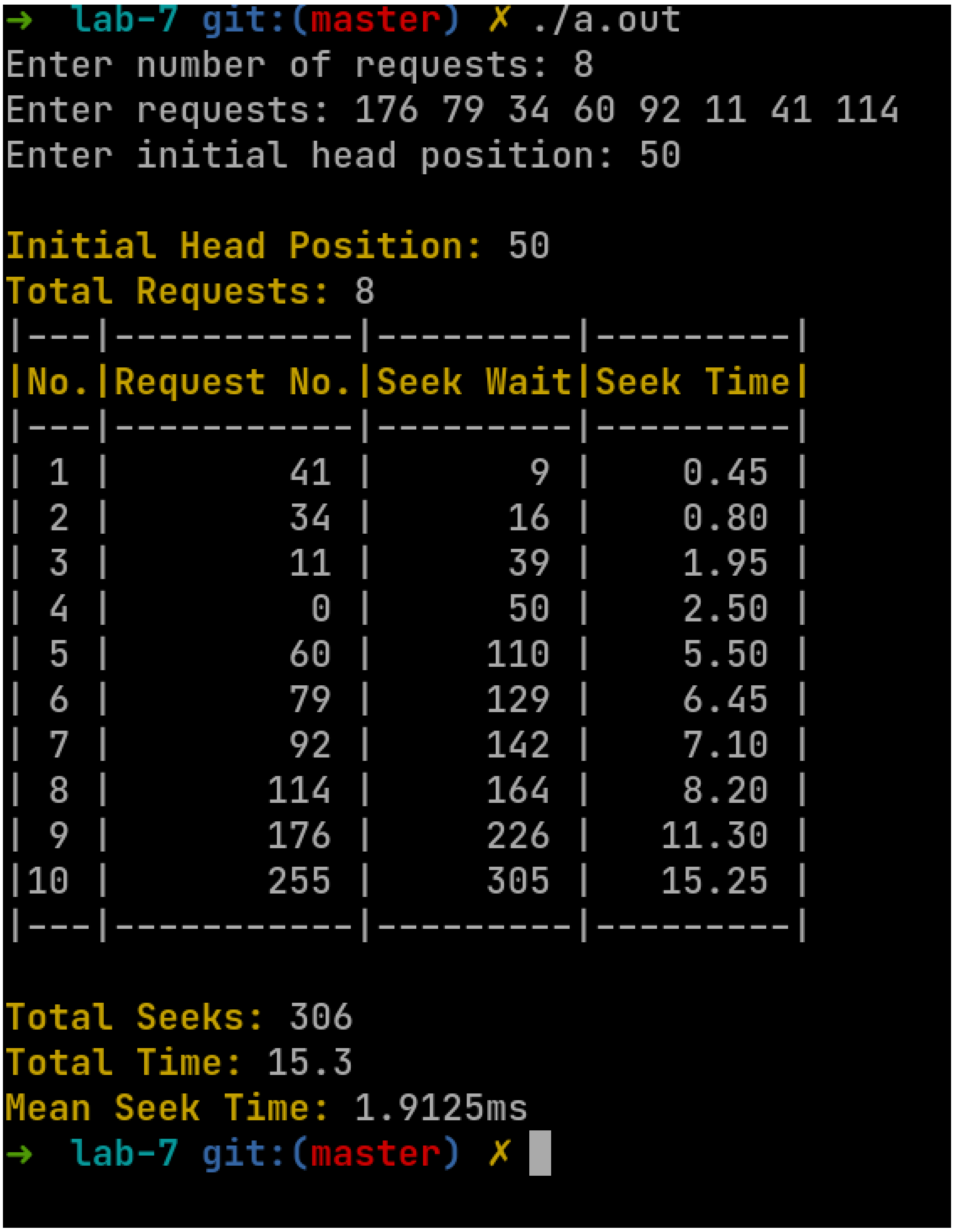
```cpp
  cout << "|---|-----------|---------|---------|\n";
  cout << BLUE_PREFIX << "|No.|Request No.|Seek Wait|Seek Time|\n"
       << BLUE_POSTFIX;
  cout << "|---|-----------|---------|---------|\n";
  int k = 1;

  while (!requests.empty()) {
    auto next = closest_to_head(requests, head);
    total_seeks += abs(head - *next);
    head = *next;

    printf("|%2d |%10d | %7d | %7.2f |\n", k++, head, total_seeks,
           total_seeks * SEEK_DELAY);
    requests.erase(next);
  }
  cout << "|---|-----------|---------|---------|\n";

  cout << endl;

  cout << BLUE_PREFIX << "Total Seeks: " << BLUE_POSTFIX << total_seeks
<< "\n";
  cout << BLUE_PREFIX << "Total Time: " << BLUE_POSTFIX
       << total_seeks * SEEK_DELAY << "\n";
  cout << BLUE_PREFIX << "Mean Seek Time: " << BLUE_POSTFIX
       << total_seeks * SEEK_DELAY / n << "ms\n";
}

int main(int argc, char const *argv[]) {
  int n;
  cout << "Enter number of requests: ";
  cin >> n;
  multiset<int> requests;
  cout << "Enter requests: ";
  for (int i = 0; i < n; ++i) {
    int temp;
    cin >> temp;
    requests.insert(temp);
  }

  int start;
  cout << "Enter initial head position: ";
  cin >> start;
  sstf(requests, start);
  return 0;
}
```



```
→  lab-7 git:(master) ✗ ./a.out
Enter number of requests: 8
Enter requests: 176 79 34 60 92 11 41 114
Enter initial head position: 50

Initial Head Position: 50
Total Requests: 8
|---|-----------|----------|----------|
|No.|Request No.|Seek Wait|Seek Time|
|---|-----------|----------|----------|
| 1 |        41 |        9 |     0.45 |
| 2 |        34 |       16 |     0.80 |
| 3 |        11 |       39 |     1.95 |
| 4 |        60 |       88 |     4.40 |
| 5 |        79 |      107 |     5.35 |
| 6 |        92 |      120 |     6.00 |
| 7 |       114 |      142 |     7.10 |
| 8 |       176 |      204 |    10.20 |
|---|-----------|----------|----------|

Total Seeks: 204
Total Time: 10.2
Mean Seek Time: 1.275ms
→  lab-7 git:(master) ✗
```

2. To implement SCAN algorithm for Disk Scheduling.

```
#include <iostream>
#include <set>
#include <vector>

using namespace std;

const int DISK_SIZE = 256;
const double SEEK_DELAY = 0.05;
const string BLUE_PREFIX = "\033[1;33m";
const string BLUE_POSTFIX = "\033[0m";

multiset<int>::iterator closest_to_head(const multiset<int> &requests,
                                        int head) {
  auto closest = requests.begin();
  int closest_distance = abs(head - *closest);

  for (auto it = requests.begin(); it != requests.end(); ++it) {
    if (abs(head - *it) < closest_distance) {
      closest_distance = abs(head - *it);
      closest = it;
    }
  }

  return closest;
}

void scan(multiset<int> &requests, int start) {
  int head = start;
  int total_seeks = 0;
  int n = requests.size();
  n -= 2;
  cout << BLUE_PREFIX << "\nInitial Head Position: " << BLUE_POSTFIX <<
head
       << "\n";

  cout << BLUE_PREFIX << "Total Requests: " << BLUE_POSTFIX << n
       << "\n";

  cout << "|---|-----------|---------|---------|\n";
  cout << BLUE_PREFIX << "|No.|Request No.|Seek Wait|Seek Time|\n"
       << BLUE_POSTFIX;
  cout << "|---|-----------|---------|---------|\n";
  int k = 1;
  for (int i = head; i >= 0; --i) {
    if (requests.count(i)) {
      printf("|%2d |%10d | %7d | %7.2f |\n", k++, i, total_seeks,
             total_seeks * SEEK_DELAY);
    }
    ++total_seeks;
    requests.erase(i);
  }

  for (int i = 1; i < DISK_SIZE; ++i) {
    if (requests.count(i)) {
      printf("|%2d |%10d | %7d | %7.2f |\n", k++, i, total_seeks,
             total_seeks * SEEK_DELAY);
    }
    ++total_seeks;
    requests.erase(i);
  }
  cout << "|---|-----------|---------|---------|\n";

  cout << endl;

  cout << BLUE_PREFIX << "Total Seeks: " << BLUE_POSTFIX << total_seeks
<< "\n";
  cout << BLUE_PREFIX << "Total Time: " << BLUE_POSTFIX
       << total_seeks * SEEK_DELAY << "\n";
  cout << BLUE_PREFIX << "Mean Seek Time: " << BLUE_POSTFIX
       << total_seeks * SEEK_DELAY / n << "ms\n";
}

int main(int argc, char const *argv[]) {
  int n;
  cout << "Enter number of requests: ";
  cin >> n;
  multiset<int> requests;
  cout << "Enter requests: ";
  for (int i = 0; i < n; ++i) {
    int temp;
    cin >> temp;
    requests.insert(temp);
  }

  requests.insert(0);
  requests.insert(DISK_SIZE - 1);

  int start;
  cout << "Enter initial head position: ";
  cin >> start;
  scan(requests, start);
  return 0;
}
```



```
➜  lab-7 git:(master) ✗ ./a.out
Enter number of requests: 8
Enter requests: 176 79 34 60 92 11 41 114
Enter initial head position: 50

Initial Head Position: 50
Total Requests: 8
|---|-----------|---------|---------|
|No.|Request No.|Seek Wait|Seek Time|
|---|-----------|---------|---------|
| 1 |        41 |       9 |    0.45 |
| 2 |        34 |      16 |    0.80 |
| 3 |        11 |      39 |    1.95 |
| 4 |         0 |      50 |    2.50 |
| 5 |        60 |     110 |    5.50 |
| 6 |        79 |     129 |    6.45 |
| 7 |        92 |     142 |    7.10 |
| 8 |       114 |     164 |    8.20 |
| 9 |       176 |     226 |   11.30 |
|10 |       255 |     305 |   15.25 |
|---|-----------|---------|---------|

Total Seeks: 306
Total Time: 15.3
Mean Seek Time: 1.9125ms
➜  lab-7 git:(master) ✗ 
```