README

1) There are 4 files

FCFS.c : “gcc FCFS.c” and “a.exe” on visual studio code or can compile on any other ide.

Input is no. of cylinder , size of request sequence ,request seq which takes cylinder no . Then arrival time for every req in input , initial position of head. Final output is First Come First Serve.

SSTF.c : “gcc SSTF.c” and “a.exe” on visual studio code or can compile on any other ide.
The arrival time is kept 0 for all the req , for sake of simplicity.
Then shortest seek time from head is given priority while scheduling.

SCAN.c : “gcc SCAN.c” and “a.exe” on visual studio code or can compile on any other ide.
Here also the arrival time is taken as 0 for simplicity.
The head will form itself to the last cylinder serving all the req in between then move backward and serve all req till the lowest cylinder number.

CSCAN.c : “gcc CSCAN.c” and “a.exe” on visual studio code or can compile on any other ide.
Here also the arrival time is taken as 0 for simplicity.
The head will form itself to the last cylinder serving all the req in between then move backward to 0 , then further move forward and server the remaianig requests.

2) so in q2 run the command “gcc q2_.c -o q2_” . Then for executing “./q2_ 32256 256”

Error Handling case : “./q2_ 32738 256”

Commands supported : _copy, _mkdir, _open, _close, _write, _chdir, _read, _rmdir