A. Design and Implementation:
    a. There are two structs, JD and partial_ans which extends JD in addition to another field. Channels store these struct variables.
    b. Sequence of initializing goroutines:
        i. Single consolidator
        ii. M workers
        iii. Single dispatcher
    c. The above sequence is followed to avoid deadlocks that might arise from unbuffered channels used.
    d. Dispatcher creates JD struct instances (jobs) and pushes them into a channel.
    e. Workers read from that channel and process the file to find the number of primes within that segment and push those results into another channel that stores the struct instance of JD along with the result generated for that particular JD.
    f. Consolidator picks from the channel into which workers write, add up those partial results and calculate the final result indicating the total number of primes present in the given binary datafile of 64-bit unsigned integers in little-endian byte order.
B. My implementation supports a maximum of 1354752 workers on my personal computer.
    a. Setting (command-line arguments) for all M, N, and C parameters:
        i. 1354752 64000 1000
        ii. Min jobs completed by a worker: 0
        iii. Max jobs completed by a worker: 1
        iv. Average jobs completed by a worker: 0.00025201612903225806
        v. Median jobs completed by a worker: 0
        vi. Elapsed time (ms): 7851 ms
C. Least elapsed time for a random data-file of 1GB in my implementation was 14907 ms.
    a. Setting (command-line arguments) for all M, N, and C parameters:
        i. 10 64000 1000
    b. Min jobs completed by a worker: 4154
    c. Max jobs completed by a worker: 4237
    d. Average jobs completed by a worker: 4194
    e. Median jobs completed by a worker: 4193
    f. Elapsed time (ms): 14907 ms
D. Largest datafile processed within 3 mins elapsed time was of size 2.6GB.
    a. Setting (command-line arguments) for all M, N, and C parameters:
        i. 10 64000 1000
        ii. Min jobs completed by a worker: 4021
        iii. Max jobs completed by a worker: 4041
        iv. Average jobs completed by a worker: 4026.6
        v. Median jobs completed by a worker: 4024

vi. Elapsed time (ms): 179855 ms

E. When the file size is large, around 2.5GB, the maximum number of supported workers can not be spawned as the number of open pointers would be too many.
   a. A file of 1.1GB with 516096 workers took 72302 ms to process.
   b. The same file with just one worker took 297046 ms.
   c. Plot of how elapsed time changes against values of M for a 1.1GB size file: