CS3510 - Operating System

Assignment 2

Soham Rajesh Pawar
CS22BTECH11055

November 27, 2023

1 Coding Approach :

The program was implemented in C to find the vampire numbers in the range 1 to `N`(obtained from the input file) by dividing the task among `K` threads.

1.1 Steps :

1. First we read `N` and `K` from the input file.
2. The main process creates `K` threads, each of which will execute a common routine.
3. According to the routine each thread will check the numberset it has been allotted to get the vampire numbers in the concerned set.
   (a) The allottment of numbers is decided using the thread id which is passed as a routine parameter(the counter variable i). Each number that is i (mod `K`) will be allotted to thread i + 1.
   (b) To check if a number is Vampire or not, we will iterate through it's factors and check each pair of factors individually.
4. If a thread identifies a pair of fangs for a number in it's set then it will record the number identified in the common log file along with it's own identity.
5. The main process waits for the completion of all threads and collects the resources.

2 Verification :

The main log file aka OutFile.log also contains the count of vampire numbers within the given range and the time required for the execution of the task for verification purposes.

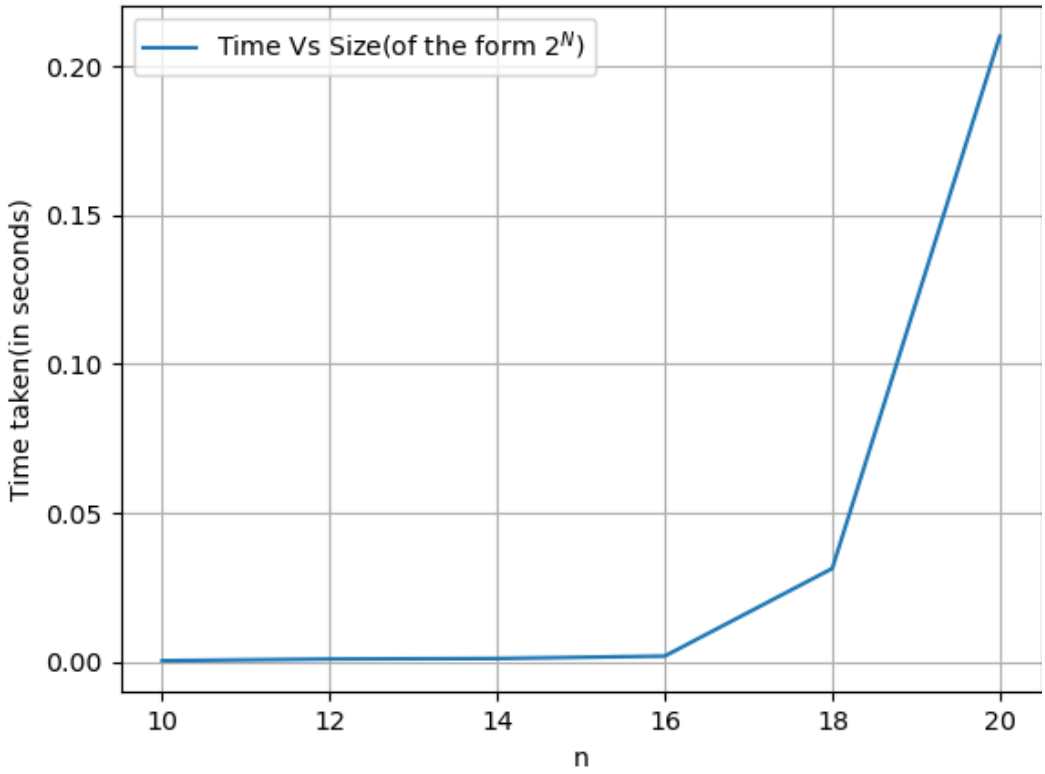
3 Conclusion :

1. Increasing the input size naturally increases the execution time.
2. Increase in the number of processes generally leads to a decrease in execution time. However, after a certain number increasing the number of processes grants diminishing returns. This could possibly be due to the distribution of vampire numbers among the natural numbers.

4 Graphs :

4.1 Maintaining K at 8

Time vs Size(2^N) :



4.2 Maintaining Size at 10,00,000

Time vs Number of Processes(K) :