**DOS Report – 1**

**(Sum Of Squares)**

Team Members:

Sai Pradyumna Reddy Chegireddy (UFID: 34631711)

Sri Greeshma Avadhootha (UFID:16136609)

Instructions to run the code :

1. To configure the desired number of workers, modify the following line 9 in Proj1.fsx

let actors = <# actors>

1. To run the project, use the following command where n is the input N and k in the input k:

dotnet fsi --langversion:preview Proj1.fsx <n> <k>

Output : On each line is the first number in the sequence for each solution.

Results :

The result of running Proj1.fsx with n = 1,000,000 and k = 4 is:

(no solution)

Running time and  **(CPU / Real)** ratio for Proj1 with n = 100,000,000 and k = 2 for different values of work unit are given in the table below.

We experiment with various work units in order to arrive at the ideal work unit. We have observed the results for the ratio of (**CPU / Real)** for a variation in the number of actors.

Utilization of all the cores along with a low real running time is desired. These metrics are important to get the ideal work unit. 100,000,000 has a relatively smaller communication time and hence we used this for analysis.

|  |  |  |  |
| --- | --- | --- | --- |
| **Work unit** | **CPU Time** | **Real time** | **Ratio** |
| 1,000 | 70.593 | 26.526 | 2.66 |
| 10,000 | 32.734 | 7.669 | 4.27 |
| 100,000 | 34.578 | 9.271 | 3.73 |
| 1,000,000 | 33.375 | 8.359 | 3.99 |
| 10,000,000 | 26.437 | 6.104 | 4.33 |

From the recorded observations in the above table, the multithreading nature of the program is comparable in all the variations because all the cores are used by the program. The ideal work unit which we have identified is 10,000. There is a trade off between core usage and number of actors. A small work unit causes the actors to complete their processing quickly and the message passing between the actors causes the overhead. On the other hand, a large work unit results in a lot of processing for each actor which becomes the bottleneck.

The largest problem we solved was :

N =100,000,000 and k = 24

Ratio = 3.048