**COP5615- Distributed Operating System Principles**

**Project 1**

**MEMBERS:**

1. Divy Nidhi Chhibber, UFID: 3138-9199

2. Wins Goyal, UFID: 7357-1559

**SIZE OF THE WORK UNIT:**

Dynamic. Depends on the input range. Maximum work per worker in one call: 10,000 numbers.

**REASONING:**

The method for allocating work to workers is dependent on the input range. For a given range, sub ranges are created. For example, for 25-60000, the subranges will be [[25 99] [1000 9999]]. These subranges are then divided equally amongst the worker. In case the divided work per worker is greater than 10000 numbers, the range is further divided equally and distributed amongst the workers. The worker is decided using a round robin fashion.

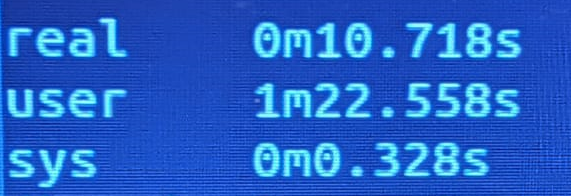
This implementation of equally dividing work is the most efficient because no worker is overworked. Also, even for bigger ranges, the maximum given a worker at a go is 10000 numbers. Increasing maximum limit above 10,000 degraded the performance.

**EXECUTING MIX RUN PROJ1.EXS 100000 200000:**

Output is in the file: Output\_mix

**RUNTIME FOR RUN PROJ1.EXS 100000 200000:**

**CPU Ratio:** 10.718/ 82.558 + 0.328 = 0.129



**LARGEST PROBLEM SOLVED:**

Largest range executed on a single system: 1-20000000.

**OBSERVER OUTPUT:**

