ASSIGNMENT 3

Name – Sarvesh Sanjay Shingare
Roll No – 144
PRN – 202101050031
Batch – B4

✷ Problem Statement : Implement the synchronization for Dining Philosopher Problem using MPI

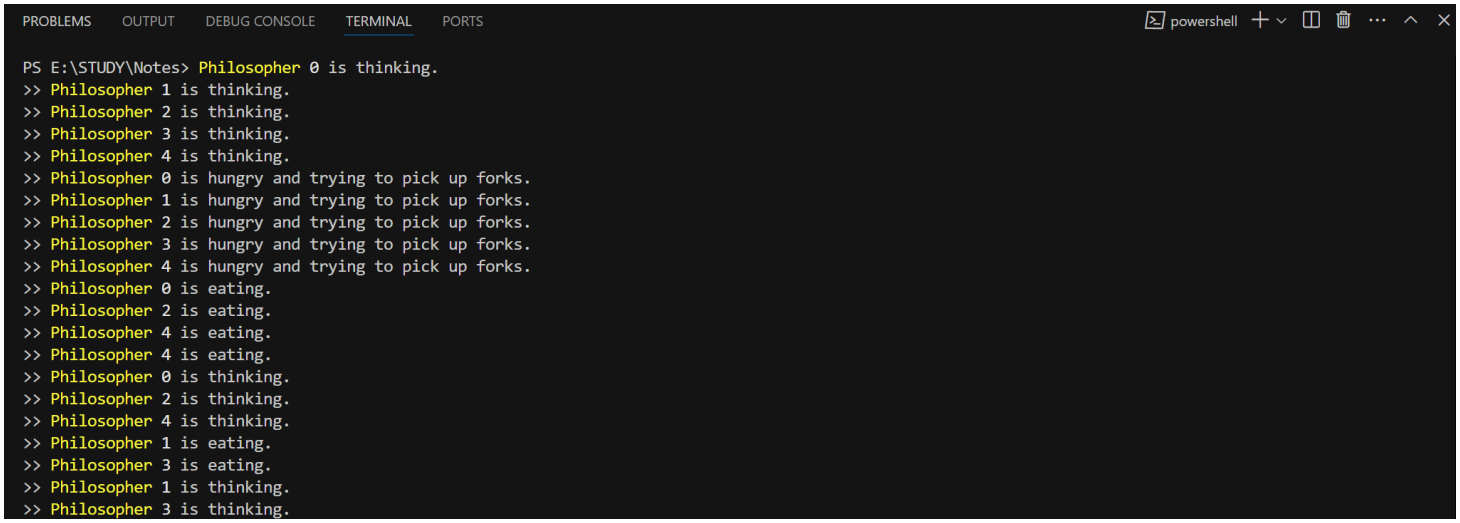
Synchronization Primitives.

Code :-

```c
// Sarvesh Shingare
// 202101050031
#include <mpi.h>
#include <stdio.h>
#include <stdlib.h>

#define THINKING 0
#define HUNGRY 1
#define EATING 2
#define LEFT (philosopher_id + num_philosophers - 1) % num_philosophers
#define RIGHT (philosopher_id + 1) % num_philosophers

void philosopher(int philosopher_id, int num_philosophers);

int main(int argc, char *argv[]) {
    int rank, num_procs;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    MPI_Comm_size(MPI_COMM_WORLD, &num_procs);

    int num_philosophers = num_procs;
    if (rank < num_philosophers) {
        philosopher(rank, num_philosophers);
    }

    MPI_Finalize();
    return 0;
}

void philosopher(int philosopher_id, int num_philosophers) {
    int state = THINKING;
    int fork_left = LEFT;
    int fork_right = RIGHT;
    MPI_Status status;

    while (1) {
        // Thinking phase
        printf("Philosopher %d is thinking.\n", philosopher_id);
```

```
        // Simulate thinking
        sleep(1);

        // Hungry phase
        state = HUNGRY;
        printf("Philosopher %d is hungry and trying to pick up forks.\n", philosopher_id);

        // Pick up the left fork
        MPI_Send(NULL, 0, MPI_INT, fork_left, 0, MPI_COMM_WORLD);
        MPI_Recv(NULL, 0, MPI_INT, fork_left, 0, MPI_COMM_WORLD, &status);

        // Pick up the right fork
        MPI_Send(NULL, 0, MPI_INT, fork_right, 0, MPI_COMM_WORLD);
        MPI_Recv(NULL, 0, MPI_INT, fork_right, 0, MPI_COMM_WORLD, &status);

        // Eating phase
        state = EATING;
        printf("Philosopher %d is eating.\n", philosopher_id);
        // Simulate eating
        sleep(1);

        // Put down the left fork
        MPI_Send(NULL, 0, MPI_INT, fork_left, 0, MPI_COMM_WORLD);

        // Put down the right fork
        MPI_Send(NULL, 0, MPI_INT, fork_right, 0, MPI_COMM_WORLD);

        // Transition back to thinking
        state = THINKING;
    }
}
```

Output :-


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
PS E:\STUDY\Notes> Philosopher 0 is thinking.
>> Philosopher 1 is thinking.
>> Philosopher 2 is thinking.
>> Philosopher 3 is thinking.
>> Philosopher 4 is thinking.
>> Philosopher 0 is hungry and trying to pick up forks.
>> Philosopher 1 is hungry and trying to pick up forks.
>> Philosopher 2 is hungry and trying to pick up forks.
>> Philosopher 3 is hungry and trying to pick up forks.
>> Philosopher 4 is hungry and trying to pick up forks.
>> Philosopher 0 is eating.
>> Philosopher 2 is eating.
>> Philosopher 4 is eating.
>> Philosopher 4 is eating.
>> Philosopher 0 is thinking.
>> Philosopher 2 is thinking.
>> Philosopher 4 is thinking.
>> Philosopher 1 is eating.
>> Philosopher 3 is eating.
>> Philosopher 1 is thinking.
>> Philosopher 3 is thinking.