COMPSCI 3SH3 Winter, 2021
Student Name: Jatin Chowdhary
Mac ID: Chowdhaj
Student #: 400033011
Date: March 5th, 2021

Lab Assignment 4 – Dining Philosophers

Q1 – Dining Philosophers

Module Description

The *dining_philosophers* program is multi-threaded C program that uses the POSIX thread library to illustrate the concurrency problem, Dining Philosophers. In this program, there are 5 philosophers, numbered 0 to 4, and there are 5 forks, numbered 0 to 4. Philosophers are represented by an array of type *pthread_t*, and forks are represented by an int array. In the *forks* array, the values can only be 1 or 0; the former signifies that a fork is available, and the latter means that the corresponding fork is not available. The program starts off by declaring and initializing preliminary stuff like threads, numbers, etc. Then, the *forks* array is populated with 1's, signifying that all forks are available. Finally, each philosopher thread is created/executed using *pthread_create* and *main* waits for the philosopher threads to finish via *pthread_join*.

Each philosopher thread executes code that corresponds to an action that a dining philosopher may do; pickup forks, eat, return forks, and think. Since the order of execution in a multi-threaded (or multi-process) program isn't guaranteed, the dining philosophers will do unpredictable things. The first thing each philosopher does is try to pickup forks via the *pickup_forks* function. It does this by checking if both left and right forks that correspond to it are available. If they are not, the philosopher will wait and try later. If the forks are available, the philosopher will pick them up, start eating, and continue eating. Subsequently, the *return_forks* function is executed, and the philosopher returns the forks and sends a signal to every other philosopher thread, indicating that forks at *X* and *Y* have been returned. Finally, the philosopher starts to think, and continues thinking for a few seconds, until the entire cycle of pickup forks, return forks, and think is executed again.

Note: This is a semi-high level description of the *dining_philosophers* program. For more information (i.e. Placement of mutexes, logic, etc.) please refer to the source file, *dining_philosophers.c*, and use the *Makefile* to compile it.

***Source Code:** *dining_philosophers.c*, *Makefile*

**Source Code is attached in a zip file*