Lab 3 – Dining Philosophers in **pthreads**

```c
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

#define N 5
#define LEFT (i + N - 1)%N
#define RIGHT (i + 1)%N
#define THINKING 0
#define HUNGRY 1
#define EATING 2

int state[N];
int threadID[N];

pthread_t tid[N];
pthread_cond_t s[N];
pthread_mutex_t mutex_lock;

void philosopher(void* p);
void think();
void eat();
void take_forks(int i);
void put_forks(int i);
void test(int i);

void philosopher(void* p) {
        int* start = (int*) p;
        int current = *start;

        for(int i=0; i<N; i++) {
                think(current);
                take_forks(current);
                eat(current);
                put_forks(current);
        }
}

void think(int i) {
        printf("Philosopher %d is thinking...\n", i+1);
        sleep(1);
}

void eat(int i) {
        printf("Philosopher %d is eating...\n", i+1);
        sleep(1);
        printf("Philosopher %d has finished eating\n", i+1);
}

//pick up forks
void take_forks(int i) {
    pthread_mutex_lock(&mutex_lock);

    state[i] = HUNGRY;
```

```c
    test(i);

    while(state[i] != EATING) {
        pthread_cond_wait(&s[i], &mutex_lock);
    }

    pthread_mutex_unlock(&mutex_lock);
}

//put down forks
void put_forks(int i) {
        pthread_mutex_lock(&mutex_lock);

    state[i] = THINKING;
    test(LEFT);
    test(RIGHT);

    pthread_mutex_unlock(&mutex_lock);
}


void test(int i) {
        if(state[i]==HUNGRY && state[LEFT]!=EATING && state[RIGHT]!=EATING) {
                state[i] = EATING;
        pthread_cond_signal(&s[i]);
        }
}

int main(int argc, char const *argv[]) {
    //initialize threads
        for(int i=0; i<N; i++) {
                state[i] = THINKING;
                threadID[i] = i;
                pthread_cond_init(&s[i], NULL);
        }

        pthread_mutex_init(&mutex_lock, NULL);

        //create philosophers
        for (int i=0; i<N; i++) {
         pthread_create(&tid[i], NULL, (void*) philosopher, (void*)
&threadID[i]);
        }

        for (int i=0; i<N; i++) {
                pthread_join(tid[i], NULL);
        }

        pthread_exit(NULL);
    return 0;
}
```

Makefile

```
CFLAGS = -Wall -g

run: lab3.o
        cc -o lab3 -pthread lab3.o

clean:
        rm lab3 lab3.o
```

Output

```
Philosopher 1 is thinking...
Philosopher 2 is thinking...
Philosopher 3 is thinking...
Philosopher 4 is thinking...
Philosopher 5 is thinking...
Philosopher 1 is eating...
Philosopher 3 is eating...
Philosopher 1 has finished eating
Philosopher 1 is thinking...
Philosopher 5 is eating...
Philosopher 3 has finished eating
Philosopher 3 is thinking...
Philosopher 2 is eating...
Philosopher 5 has finished eating
Philosopher 5 is thinking...
Philosopher 4 is eating...
Philosopher 2 has finished eating
Philosopher 2 is thinking...
Philosopher 1 is eating...
Philosopher 4 has finished eating
Philosopher 4 is thinking...
Philosopher 3 is eating...
Philosopher 1 has finished eating
Philosopher 1 is thinking...
Philosopher 5 is eating...
Philosopher 3 has finished eating
Philosopher 3 is thinking...
Philosopher 2 is eating...
Philosopher 5 has finished eating
Philosopher 5 is thinking...
```

```
Philosopher 4 is eating...
Philosopher 2 has finished eating
Philosopher 2 is thinking...
Philosopher 1 is eating...
Philosopher 4 has finished eating
Philosopher 4 is thinking...
Philosopher 3 is eating...
Philosopher 1 has finished eating
Philosopher 1 is thinking...
Philosopher 5 is eating...
Philosopher 3 has finished eating
Philosopher 3 is thinking...
Philosopher 2 is eating...
Philosopher 5 has finished eating
Philosopher 5 is thinking...
Philosopher 4 is eating...
Philosopher 2 has finished eating
Philosopher 2 is thinking...
Philosopher 1 is eating...
Philosopher 4 has finished eating
Philosopher 4 is thinking...
Philosopher 3 is eating...
Philosopher 1 has finished eating
Philosopher 1 is thinking...
Philosopher 5 is eating...
Philosopher 3 has finished eating
Philosopher 3 is thinking...
Philosopher 2 is eating...
Philosopher 5 has finished eating
Philosopher 5 is thinking...
```

```
Philosopher 4 is eating...
Philosopher 2 has finished eating
Philosopher 2 is thinking...
Philosopher 1 is eating...
Philosopher 4 has finished eating
Philosopher 4 is thinking...
Philosopher 3 is eating...
Philosopher 1 has finished eating
Philosopher 5 is eating...
Philosopher 3 has finished eating
Philosopher 2 is eating...
Philosopher 5 has finished eating
Philosopher 4 is eating...
Philosopher 2 has finished eating
Philosopher 4 has finished eating
```