Arti bhatia

CS5A,16

Q- simulate dining philo prob

// Dining philosopher's problem simulation using semaphores

#include <stdio.h>

#include <stdlib.h>

#include <pthread.h>

#include <semaphore.h>

#include <unistd.h>

sem\_t chopstick[5];

void \*philos(void \*);

void eat(int);

int main()

{

int i, n[5];

pthread\_t T[5];

for (int i = 0; i < 5; i++)

{

sem\_init(&chopstick[i], 0, 1);

}

for (int i = 0; i < 5; i++)

{

n[i] = i;

pthread\_create(&T[i], NULL, philos, (void \*)&n[i]); // creating a thread for eachphilosopher

}

for (int i = 0; i < 5; i++)

{

pthread\_join(T[i], NULL);

}

}

void \*philos(void \*n)

{

int ph = \*(int \*)n;

printf("\nPhilosopher %d is Hungry\n\n", ph);

printf("Philosopher %d tries to pick the left chopstick\n\n", ph);

sem\_wait(&chopstick[ph]);

printf("Philosopher %d takes the left chopstick\n\n", ph);

printf("Philosopher %d tries to pick the right chopstick\n\n", ph);

sem\_wait(&chopstick[(ph + 1) % 5]);

printf("Philosopher %d takes the right chopstick\n\n", ph);

eat(ph);

sleep(2);

printf("Philosopher %d finished eating\n\n", ph);

sem\_post(&chopstick[(ph + 1) % 5]);

printf("Philosopher %d has put down the right chopstick\n\n", ph);

sem\_post(&chopstick[ph]);

printf("Philosopher %d has put down the left chopstick\n", ph);

}

void eat(int ph)

{

printf("Philosopher %d begins to eat\n\n", ph);

}

