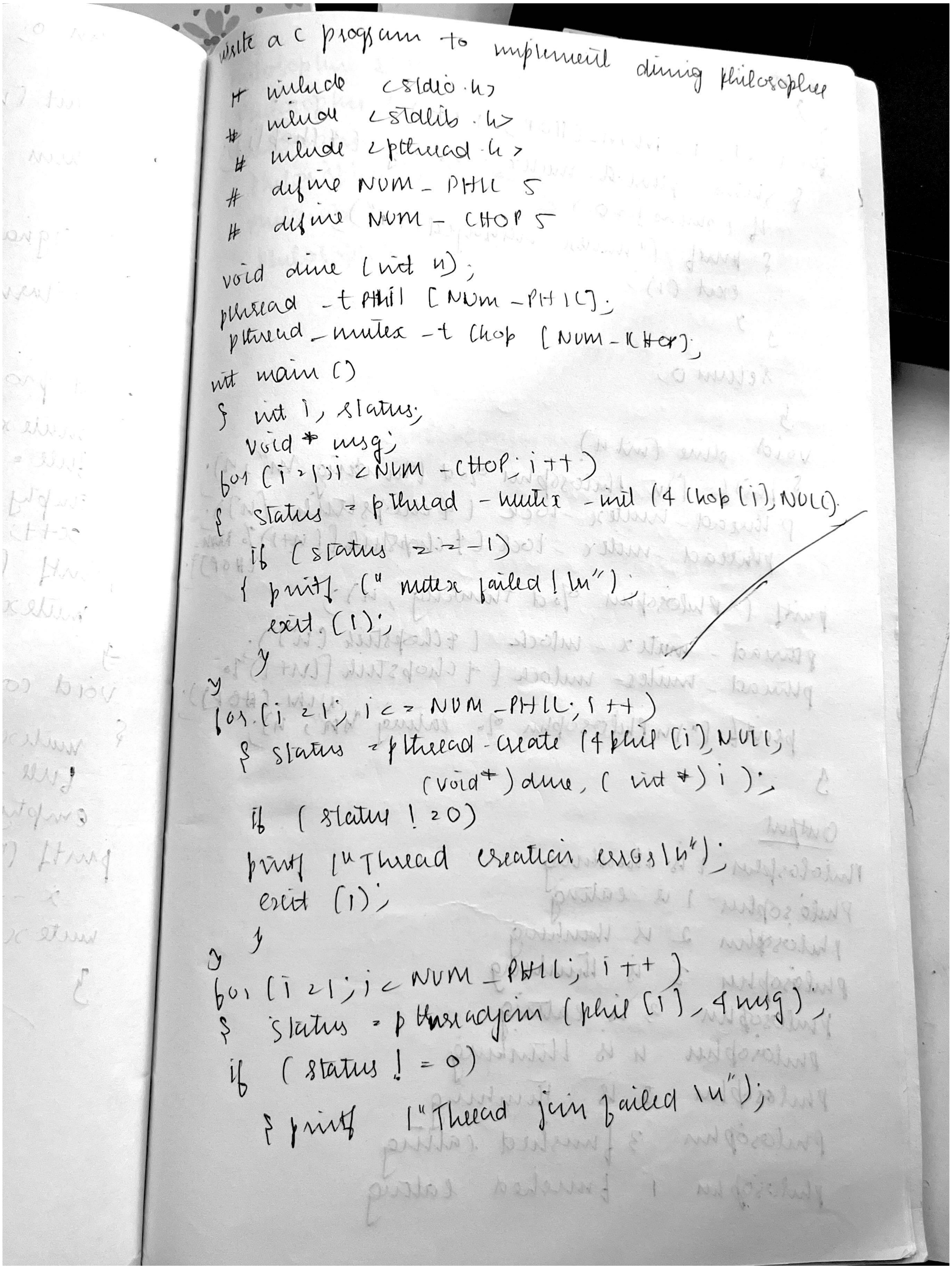
Write a C program to simulate the concept of Dining-Philosophers problem.



write a C program to implement dining philosopher

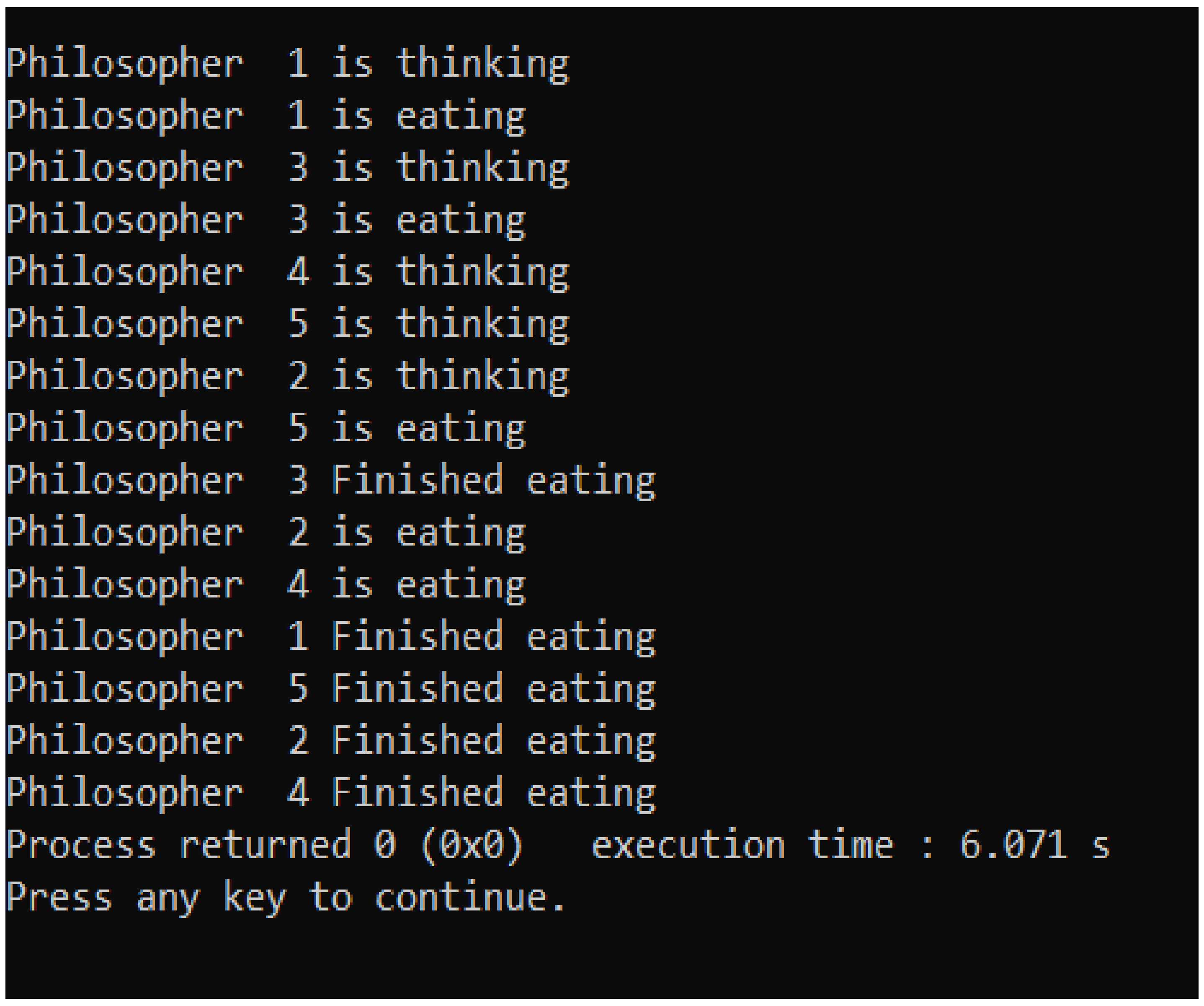
```
# include <stdio.h>
# include <stdlib.h>
# include <pthread.h>
# define NUM_PHIL 5
# define NUM_CHOP 5

void dine (int n);
pthread_t phil [NUM_PHIL];
pthread_mutex_t chop [NUM_CHOP];
int main ()
{ int i, status;
  void * msg;
for (i=1; i<NUM_CHOP; i++)
{ status = pthread_mutex_init (&chop[i], NULL).
  if (status ==-1)
  { printf (" mutex failed! \n");
    exit (1);
  }
}
for (i=1; i<= NUM_PHIL; i++)
  { status = pthread_create (&phil[i], NULL,
                 (void*) dine, (int *) i );
   if (status != 0)
   printf ("Thread creation error \n");
    exit (1);
    }
}
for (i=1; i<NUM_PHIL; i++)
{ status = pthreadjoin (phil[i], &msg);
 if (status != 0)
   { printf ("Thread join failed \n");
exit (1);
  }
}
for (i=1; i<NUM_CHOP; i++)
  { status = pthread_mutex_distroy (&chop[i]);
     if (status != 0)
     { printf ("mutex destroyed!\n");
        exit (1);
     }
   }
     return 0;
  }
  void   dine (int n)
  { printf ("\n Philosopher %d Ithinking \n", n);
    pthread_mutex_lock (&chopstick [n]);
     pthread_mutex_lock (&chopstick [(n+1) % NUM_CHOP]
 printf ("Philosopher %d thinking", n);
  pthread_mutex_unlock (&chopstick [n]);
  pthread_mutex_unlock (&chopstick [(n+1) % NUM_CHOP]
   printf ("\n Philosopher % eating \n", n);
  }
```

Output

```
Philosopher 1 is thinking
 Philosopher 1 is eating
  philosopher 2 is thinking
 philosopher 3 is thinking
  philosopher 3 is eating
   philosopher 4 is thinking
   philosopher 5 is thinking
   philosopher 3 finished eating
   philosopher 1 finished eating
```

```
Philosopher 2 is eating
Philosopher 4 is eating
Philosopher 5 is eating
Philosopher 2 finished eating
Philosopher 5 finished eating
Philosopher 4 finished eating
```



```
Philosopher  1 is thinking
Philosopher  1 is eating
Philosopher  3 is thinking
Philosopher  3 is eating
Philosopher  4 is thinking
Philosopher  5 is thinking
Philosopher  2 is thinking
Philosopher  5 is eating
Philosopher  3 Finished eating
Philosopher  2 is eating
Philosopher  4 is eating
Philosopher  1 Finished eating
Philosopher  5 Finished eating
Philosopher  2 Finished eating
Philosopher  4 Finished eating
Process returned 0 (0x0)   execution time : 6.071 s
Press any key to continue.
```