

1. การทำงานของโปรแกรมนี้ จะสร้าง thread ขึ้นมา 5 thread ทำหน้าที่หน้าที่เป็น worker

โดยทำหน้าที่ AddButter, AddEgg, Heatbowl, Eater

หลักการทำงานคือ ไข่เนยและไข่ก่อน แล้วนำไปอบ แล้วจึงกิน (เนยและไข่สามารถใส่พร้อมกันได้)

กลไกการบังคับอยู่ที่ while loop ถ้าเค้กพร้อมอบ/พร้อมกินอยู่แล้ว จะให้ cond_wait รอไปก่อน แต่ถ้าเค้กยังไม่พร้อมอบ/พร้อมกิน ก็จะทำให้ทำการใส่ส่วนผสม/อบ และซิกนัลต่อไป

2. Pthread_cond_wait จะเป็นตัวปลดล็อค lock และนำ condition ช่างหน้าไปเข้าคิว ทำให้ไม่เกิด dead lock

3.

```
void Chassis() {
    Acq lock
    while(1){
        while(wait_condition){
            wait(chassis)
        }
    }
    add Chassis
    broadcast(assembly)
}
```

```
void Seats, Engine, Tires() {
    Acq lock
    while(1){
        while(wait_condition){
            wait(assembly)
        }
    }
    add Seats, Engine, Tires
    signal(cover)
}
```

```
void Cover() {
    Acq lock
    while(1){
        while(wait_condition){
            wait(cover)
        }
    }
    add Cover
    signal(paint)
}
```

```

void Paint() {
    Acq lock
    while(1){
        while(wait_condition){
            wait(paint)
        }
    }
    Paint
    signal(Chassis)
}

```

```

Void worker_a() {
    Acq lock
    while(1){
        while(wait_condition)
            wait(this_wok)
        Work1()
        signal(next_work)

        while(wait_condition2)
            wait(this_work)
        Work2()
        signal(next_work)
    }
}

```

```

Void worker_b(){
    Same as Chassis function
}

```

```

Void worker_c(){
    Same as Seats
}

```

```

Void worker_d() {
    Acq lock
    while(1){
        while(wait_condition)
            wait(this_work)
        Work1
        signal(next_work)

        while(wait_condition2)
            wait(this_work)
        Work2()
        signal(next_work)
    }
}

```

Chassis -> Tires, Seats, Engine -> cover -> Paint
 Mutex : chassis, assembly, cover, paint, lock