Final Report CSE344 Eren Torlak 210104004090

How to run ?

make //compiles

make run-server

make run-client

you can use this to get ip : ip addr show eth0 | grep

-oP '(?<=inet\s)\d+(\.\d+){3}'

Server Side :

It gets 4 argument : [portnumber] [CookthreadPoolSize] [DeliveryPoolSize] [k]

Log file is opened then socket is initilized with necessary permissions.

Cook and delivery threads are initiated then with pthread create created.

```
    pthread_t cook_threads[MAX_COOKS];
    pthread_t delivery_threads[MAX_DELIVERIES];
```

And one more thread for managing and dispatching. It is client_handler thread.

Client_handler thread:

It does get location of clients and orders.

If -1 -1 is comes it cancels the order.

Cook thread:

It does preps or cooks while shop is open.

It does use mutexes for race conditions.

Prepration is done with two psodo_inverse function .

If an order cancelled it does notify client.

After prepration to simulate oven usage we use mutex and cond variable.

```
// Simulate oven usage
pthread_mutex_lock(&oven_mutex);
while (available_oven_aparatus <= 0 || available_oven_capacity <= 0) {
    pthread_cond_wait(&oven_cond, &oven_mutex);
}
available_oven_aparatus--;
available_oven_capacity--;
pthread_mutex_unlock(&oven_mutex);
```

If order is cancelled oven operatus is released.

After cooking it notifys client.

And then releases aparatus and signals delivery person.

Delivery Thread :

If shop is closed then it notifys client then closes socket.

If shop is open it is in loop .

It does necessary notfying such as if cancellation happens when delivering it notifys client.

And also it waits until bags are full.

İf last remainder tasks cant full the bag delivery person carrys them one by one.

According to distance of location sleep is done.

Each time delivery person transports pide its amount of delivery score is incremented.

Handle Signal :

When cancel signal came from server it does make cancelled=1 for each order that is not delivered.

And makes shop_open variable 0 .

It is then notfyied to client and clients are cancelled accordingly.

Client Side:

It generates client threads.

Each client thread connects server with socket.

It does sent server its random location.

If “order delivered “notification cames from server it does ends itself normally.it notifys server about that.

Handle_signal client side :

It does make cancel_all=1 . this makes every client thread to notfiy server about cancellation of order.

Logging:

In Each important step everything is logged in txt file. Output is in last images.

Final Report CSE344 Eren Torlak 210104004090

Edge Cases :

1-) server ctrl + c:

Cook case:

```
Client at (2, 16): You are Client id 14 , your order was cancelled during cooking by cook 1
Client at (1, 8): You are Client id 15 , your order was cancelled during cooking by cook 4
Client at (7, 9): You are Client id 16 , your order was cancelled during cooking by cook 3
```

Prep case:

```
Client at (0, 19): Your order was cancelled during preparation.
Client at (7, 15): Your order was cancelled during preparation.
Client at (6, 12): Your order was cancelled during preparation.
```

Delivery case:

```
Order 3 was cancelled while in delivery by delivery person 4
Order 5 was cancelled while in delivery by delivery person 4
Order 6 was cancelled while in delivery by delivery person 4
```

2-) client ctrl + c :

Client side :

```
Client at (7, 15): You are Client id 6 , your order is out for delivery by delivery person 1
Client at (2, 16): You are Client id 14 , your order is out for delivery by delivery person 1
Client at (2, 16): You are Client id 14 , your order is cooked by cook 4
Client at (6, 12): You are Client id 8 , your order is prepared by cook 4
^CClient shutting down
Client at (7, 9) ,  client socket: 14 disconnecting
Cancellation message sent
Client at (3, 6) ,  client socket: 10 disconnecting
Cancellation message sent
Client at (6, 13) ,  client socket: 3 disconnecting
Cancellation message sent
Client 0 joined
Client at (0, 6) ,  client socket: 11 disconnecting
Cancellation message sent
Client at (1, 8) ,  client socket: 13 disconnecting
Cancellation message sent
Client at (0, 19) ,  client socket: 9 disconnecting
Cancellation message sent
Client at (9, 1) ,  client socket: 7 disconnecting
Cancellation message sent
Client at (3, 15) ,  client socket: 5 disconnecting
Cancellation message sent
Client at (2, 7) ,  client socket: 8 disconnecting
Cancellation message sent
Client at (6, 12) ,  client socket: 6 disconnecting
Cancellation message sent
Client at (7, 15) ,  client socket: 4 disconnecting
Cancellation message sent
Client 1 joined
Client 2 joined
Client 3 joined
Client 4 joined
Client 5 joined
Client 6 joined
Client 7 joined
Client 8 joined
Client at (2, 10) ,  client socket: 15 disconnecting
Cancellation message sent
Client at (2, 16) ,  client socket: 12 disconnecting
Cancellation message sent
Client 9 joined
Client 10 joined
Client 11 joined
Client 12 joined
```

Final Report CSE344 Eren Torlak 210104004090

Server Side:

```
/PideShop 8099 4 6 1
lient connected at (6, 13)  , client socket: 5  , order id : 1
lient connected at (7, 15)  , client socket: 6  , order id : 2
lient connected at (7, 9)  , client socket: 16  , order id : 3
lient connected at (2, 16)  , client socket: 14  , order id : 4
lient connected at (0, 19)  , client socket: 11  , order id : 5
lient connected at (3, 15)  , client socket: 7  , order id : 6
lient connected at (6, 12)  , client socket: 8  , order id : 7
lient connected at (3, 6)  , client socket: 12  , order id : 8
lient connected at (0, 6)  , client socket: 13  , order id : 9
lient connected at (1, 8)  , client socket: 15  , order id : 10
lient connected at (9, 1)  , client socket: 9  , order id : 11
lient connected at (2, 7)  , client socket: 10  , order id : 12
lient connected at (2, 10)  , client socket: 17  , order id : 13
lient disconnected. client socket: 16
lient disconnected. client socket: 12
lient disconnected. client socket: 5
lient disconnected. client socket: 13
lient disconnected. client socket: 15
lient disconnected. client socket: 11
lient disconnected. client socket: 9
lient disconnected. client socket: 7
lient disconnected. client socket: 10
lient disconnected. client socket: 8
lient disconnected. client socket: 6
lient disconnected. client socket: 17
lient disconnected. client socket: 14
```

3-) When a client connected and ended but server still runs another client can connect again flawlessly.

After disconnecions new connects are came:

```
Client disconnected. client socket: 38
Client disconnected. client socket: 45
Client disconnected. client socket: 47
Client disconnected. client socket: 48
Client disconnected. client socket: 51
Client disconnected. client socket: 46
Client disconnected. client socket: 53
Client disconnected. client socket: 20
Client disconnected. client socket: 49
Client disconnected. client socket: 21
Client disconnected. client socket: 52
Client connected at (3, 6)  , client socket: 55  , order id : 51
Client connected at (3, 15)  , client socket: 6  , order id : 52
Client connected at (6, 12)  , client socket: 7  , order id : 53
Client connected at (1, 8)  , client socket: 14  , order id : 54
Client connected at (9, 2)  , client socket: 19  , order id : 55
Client connected at (7, 15)  , client socket: 5  , order id : 56
Client connected at (0, 19)  , client socket: 10  , order id : 57
Client connected at (3, 6)  , client socket: 11  , order id : 58
Client connected at (0, 6)  , client socket: 12  , order id : 59
Client connected at (2, 16)  , client socket: 13  , order id : 60
Client connected at (9, 1)  , client socket: 8  , order id : 61
Client connected at (8, 4)  , client socket: 46  , order id : 62
Client connected at (2, 10)  , client socket: 16  , order id : 63
Client connected at (2, 3)  , client socket: 17  , order id : 64
Client connected at (7, 15)  , client socket: 18  , order id : 65
Client connected at (2, 7)  , client socket: 9  , order id : 66
Client connected at (2, 18)  , client socket: 20  , order id : 67
Client connected at (9, 7)  , client socket: 21  , order id : 68
Client connected at (3, 16)  , client socket: 22  , order id : 69
Client connected at (1, 2)  , client socket: 23  , order id : 70
Client connected at (9, 13)  , client socket: 24  , order id : 71
```

4-> when there is a case that bag is not full but there are no other orders left. My code handles this problem.Delivery person delivers one by one if bag is not full in the end of program.

```
Order 10 added to delivery person 6's bag
Order 11 added to delivery person 6's bag
Order 5 delivered by delivery person 4
Order 12 is cooked by cook 2
Order 12 added to delivery person 6's bag
Delivery person 6 is delivering orders
Order 13 is prepared by cook 3
Order 14 is prepared by cook 4
Order 6 delivered by delivery person 4
Order 7 delivered by delivery person 5
Order 13 is cooked by cook 3
Order 13 added to delivery person 1's bag
Delivery person 1 is delivering orders
Order 14 is cooked by cook 4
Order 14 added to delivery person 2's bag
Delivery person 2 is delivering orders
Order 14 delivered by delivery person 2
Order 8 delivered by delivery person 5
Order 10 delivered by delivery person 6
Order 9 delivered by delivery person 5
Order 11 delivered by delivery person 6
Order 13 delivered by delivery person 1
Order 12 delivered by delivery person 6
```

Final Report CSE344 Eren Torlak 210104004090

Delivery person 6s bag is 3 so full.

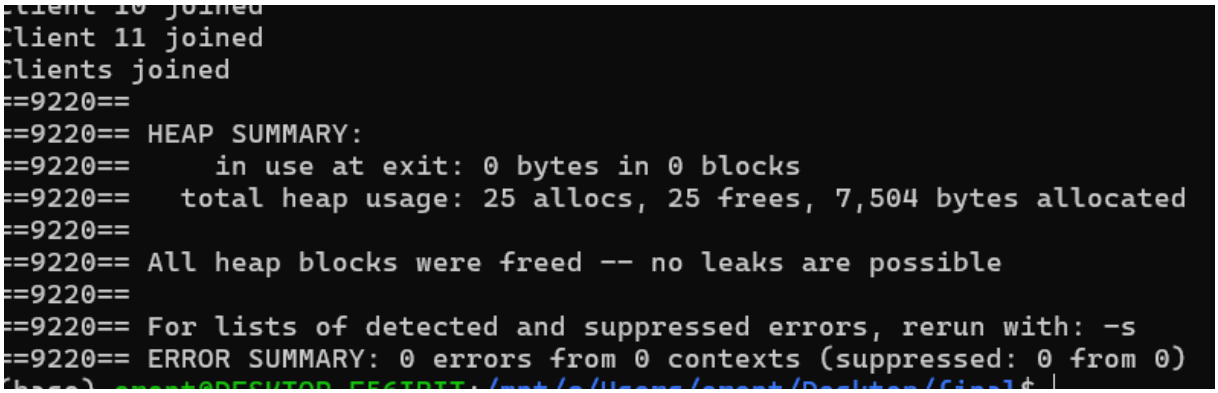
Thera are two more order left .

One of them is handled by delivery person 1 and other one by delivery person 2. They immediately goes to delivery without waiitng because there is no one to wait for.

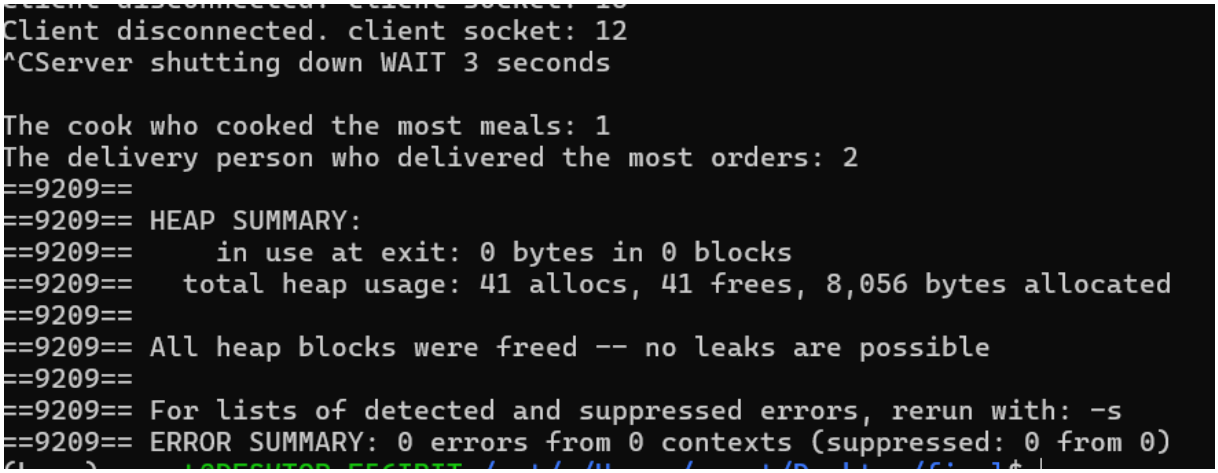
Final Report CSE344 Eren Torlak 210104004090

5-> Memory leak test :

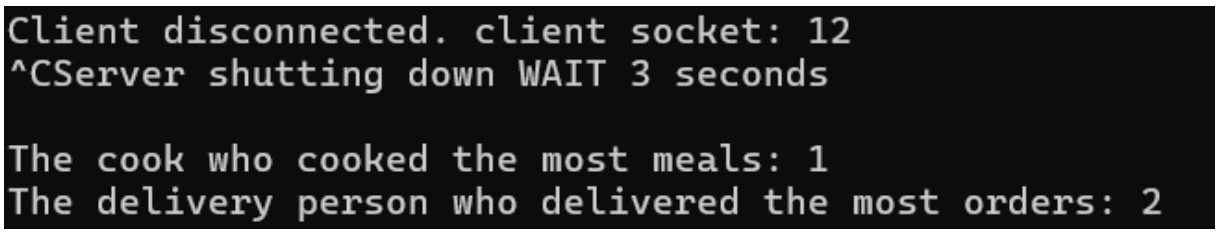
Client side :



Server side :



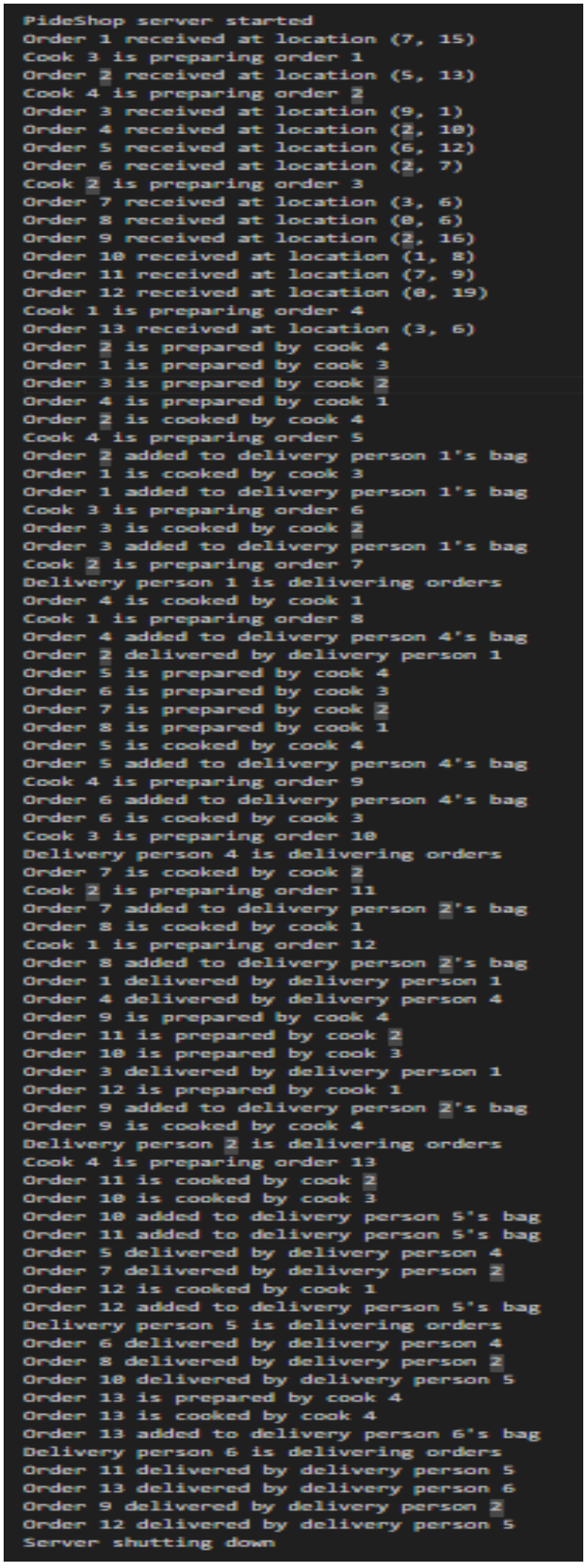
6-> the best cook and delivery person is promoted :



One of the unique features of my code is if clients order is delivered then thread finishes his job , disconnects server.

Final Report CSE344 Eren Torlak 210104004090

Log file :



```
PideShop server started
Order 1 received at location (7, 15)
Cook 3 is preparing order 1
Order 2 received at location (5, 13)
Cook 4 is preparing order 2
Order 3 received at location (9, 1)
Order 4 received at location (2, 10)
Order 5 received at location (6, 12)
Order 6 received at location (2, 7)
Cook 2 is preparing order 3
Order 7 received at location (3, 6)
Order 8 received at location (0, 6)
Order 9 received at location (2, 16)
Order 10 received at location (1, 8)
Order 11 received at location (7, 9)
Order 12 received at location (0, 19)
Cook 1 is preparing order 4
Order 13 received at location (3, 6)
Order 2 is prepared by cook 4
Order 1 is prepared by cook 3
Order 3 is prepared by cook 2
Order 4 is prepared by cook 1
Order 2 is cooked by cook 4
Cook 4 is preparing order 5
Order 2 added to delivery person 1's bag
Order 1 is cooked by cook 3
Order 1 added to delivery person 1's bag
Cook 3 is preparing order 6
Order 3 is cooked by cook 2
Order 3 added to delivery person 1's bag
Cook 2 is preparing order 7
Delivery person 1 is delivering orders
Order 4 is cooked by cook 1
Cook 1 is preparing order 8
Order 4 added to delivery person 4's bag
Order 2 delivered by delivery person 1
Order 5 is prepared by cook 4
Order 6 is prepared by cook 3
Order 7 is prepared by cook 2
Order 8 is prepared by cook 1
Order 5 is cooked by cook 4
Order 5 added to delivery person 4's bag
Cook 4 is preparing order 9
Order 6 added to delivery person 4's bag
Order 6 is cooked by cook 3
Cook 3 is preparing order 10
Delivery person 4 is delivering orders
Order 7 is cooked by cook 2
Cook 2 is preparing order 11
Order 7 added to delivery person 2's bag
Order 8 is cooked by cook 1
Cook 1 is preparing order 12
Order 8 added to delivery person 2's bag
Order 1 delivered by delivery person 1
Order 4 delivered by delivery person 4
Order 9 is prepared by cook 4
Order 11 is prepared by cook 2
Order 10 is prepared by cook 3
Order 3 delivered by delivery person 1
Order 12 is prepared by cook 1
Order 9 added to delivery person 2's bag
Order 9 is cooked by cook 4
Delivery person 2 is delivering orders
Cook 4 is preparing order 13
Order 11 is cooked by cook 2
Order 10 is cooked by cook 3
Order 10 added to delivery person 5's bag
Order 11 added to delivery person 5's bag
Order 5 delivered by delivery person 4
Order 7 delivered by delivery person 2
Order 12 is cooked by cook 1
Order 12 added to delivery person 5's bag
Delivery person 5 is delivering orders
Order 6 delivered by delivery person 4
Order 8 delivered by delivery person 2
Order 10 delivered by delivery person 5
Order 13 is prepared by cook 4
Order 13 is cooked by cook 4
Order 13 added to delivery person 6's bag
Delivery person 6 is delivering orders
Order 11 delivered by delivery person 5
Order 13 delivered by delivery person 6
Order 9 delivered by delivery person 2
Order 12 delivered by delivery person 5
Server shutting down
```

Final Report CSE344 Eren Torlak 210104004090

Server side Output :

```
(base) erent@DESKTOP-E56IRIT:/mnt/c/Users/erent/Desktop/final$ make do
./PideShop 8099 4 6 1
Client connected at (7, 15)  , client socket: 5  , order id : 1
Client connected at (3, 6)  , client socket: 6  , order id : 2
Client connected at (9, 1)  , client socket: 9  , order id : 3
Client connected at (2, 16)  , client socket: 14  , order id : 4
Client connected at (2, 10)  , client socket: 17  , order id : 5
Client connected at (3, 15)  , client socket: 7  , order id : 6
Client connected at (6, 12)  , client socket: 8  , order id : 7
Client connected at (0, 6)  , client socket: 13  , order id : 8
Client connected at (2, 7)  , client socket: 10  , order id : 9
Client connected at (0, 19)  , client socket: 11  , order id : 10
Client connected at (1, 8)  , client socket: 15  , order id : 11
Client connected at (3, 6)  , client socket: 12  , order id : 12
Client connected at (7, 9)  , client socket: 16  , order id : 13
Client disconnected. client socket: 5
Client disconnected. client socket: 6
Client disconnected. client socket: 9
Client disconnected. client socket: 14
Client disconnected. client socket: 17
Client disconnected. client socket: 8
Client disconnected. client socket: 13
Client disconnected. client socket: 10
Client disconnected. client socket: 11
Client disconnected. client socket: 7
Client disconnected. client socket: 15
Client disconnected. client socket: 16
Client disconnected. client socket: 12
^CServer shutting down WAIT 3 seconds

The cook who cooked the most meals: 3
The delivery person who delivered the most orders: 1
```

Final Report CSE344 Eren Torlak 210104004090

Client Side output:

```
Clients joined
(base) erent@DESKTOP-E56IRIT:/mnt/c/Users/erent/Desktop/final$ make run-client
./HungryVeryMuch 127.0.0.1 8099 13 10 20
Client connected at (3, 6) to server 127.0.0.1:8099
Client connected at (9, 1) to server 127.0.0.1:8099
Client connected at (7, 15) to server 127.0.0.1:8099
Client connected at (2, 7) to server 127.0.0.1:8099
Client connected at (6, 12) to server 127.0.0.1:8099
Client connected at (2, 16) to server 127.0.0.1:8099
Client connected at (3, 15) to server 127.0.0.1:8099
Client connected at (0, 19) to server 127.0.0.1:8099
Client connected at (3, 6) to server 127.0.0.1:8099
Client connected at (0, 6) to server 127.0.0.1:8099
Client connected at (1, 8) to server 127.0.0.1:8099
Client connected at (7, 9) to server 127.0.0.1:8099
Client connected at (2, 10) to server 127.0.0.1:8099
Client at (7, 15): You are Client id 5 , your order is received
Client at (7, 15): You are Client id 5 , your order is prepared by cook 3
Client at (3, 6): You are Client id 6 , your order is received
Client at (3, 6): You are Client id 6 , your order is prepared by cook 4
Client at (9, 1): You are Client id 9 , your order is received
Client at (2, 16): You are Client id 14 , your order is received
Client at (2, 10): You are Client id 17 , your order is received
Client at (9, 1): You are Client id 9 , your order is prepared by cook 1
Client at (3, 15): You are Client id 7 , your order is received
Client at (6, 12): You are Client id 8 , your order is received
Client at (0, 6): You are Client id 13 , your order is received
Client at (2, 7): You are Client id 10 , your order is received
Client at (0, 19): You are Client id 11 , your order is received
Client at (1, 8): You are Client id 15 , your order is received
Client at (2, 16): You are Client id 14 , your order is prepared by cook 2
Client at (3, 6): You are Client id 12 , your order is received
Client at (7, 9): You are Client id 16 , your order is received
Client at (7, 15): You are Client id 5 , your order is cooked by cook 3
Client at (2, 10): You are Client id 17 , your order is prepared by cook 3
Client at (9, 1): You are Client id 9 , your order is cooked by cook 1
Client at (3, 15): You are Client id 7 , your order is prepared by cook 1
Client at (7, 15): You are Client id 5 , your order is out for delivery by delivery person 1
Client at (3, 6): You are Client id 6 , your order is cooked by cook 4
Client at (3, 6): You are Client id 6 , your order is out for delivery by delivery person 1
Client at (9, 1): You are Client id 9 , your order is out for delivery by delivery person 1
Client at (6, 12): You are Client id 8 , your order is prepared by cook 4
Client at (2, 16): You are Client id 14 , your order is cooked by cook 2
Client at (0, 6): You are Client id 13 , your order is prepared by cook 2
Client at (2, 16): You are Client id 14 , your order is out for delivery by delivery person 4
Client at (7, 15): Order delivered
Client at (2, 7): You are Client id 10 , your order is prepared by cook 3
Client at (2, 10): You are Client id 17 , your order is out for delivery by delivery person 4
Client at (3, 15): You are Client id 7 , your order is out for delivery by delivery person 4
Client at (0, 19): You are Client id 11 , your order is prepared by cook 1
Client at (1, 8): You are Client id 15 , your order is prepared by cook 4
Client at (6, 12): You are Client id 8 , your order is out for delivery by delivery person 2
Client at (3, 6): Order delivered
Client at (3, 6): You are Client id 12 , your order is prepared by cook 2
Client at (0, 6): You are Client id 13 , your order is out for delivery by delivery person 2
Client at (9, 1): Order delivered
Client at (2, 16): Order delivered
Client at (7, 9): You are Client id 16 , your order is prepared by cook 3
Client at (2, 7): You are Client id 10 , your order is out for delivery by delivery person 2
Client at (0, 19): You are Client id 11 , your order is out for delivery by delivery person 6
Client at (1, 8): You are Client id 15 , your order is out for delivery by delivery person 6
Client at (3, 6): You are Client id 12 , your order is out for delivery by delivery person 6
Client at (2, 10): Order delivered
Client at (6, 12): Order delivered
Client at (0, 6): Order delivered
Client at (7, 9): You are Client id 16 , your order is out for delivery by delivery person 5
Client at (2, 7): Order delivered
Client at (0, 19): Order delivered
Client at (3, 15): Order delivered
Client 0 joined
Client 1 joined
Client 2 joined
Client 3 joined
Client 4 joined
Client at (1, 8): Order delivered
Client at (7, 9): Order delivered
Client at (3, 6): Order delivered
Client 5 joined
Client 6 joined
Client 7 joined
Client 8 joined
Client 9 joined
Client 10 joined
Client 11 joined
Client 12 joined
Clients joined
```