IERG3050\_2021 Spring\_Assignment 4

Name: To Yi Him SID: 1155126366

**Part A,B,C**

All outputs are shown in the folder.

Config.csv can modify the input.

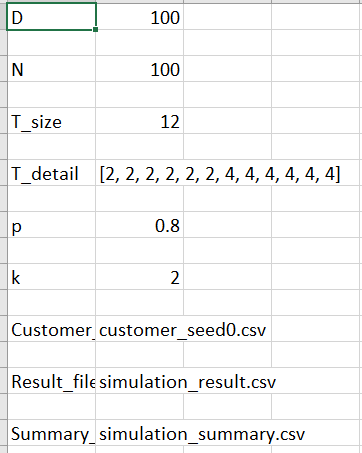
Most of the explanation of program code are commented inside the program: assignment4\_basic.py

All functions required work properly, and most of the output are similar to the sample result under same seed.

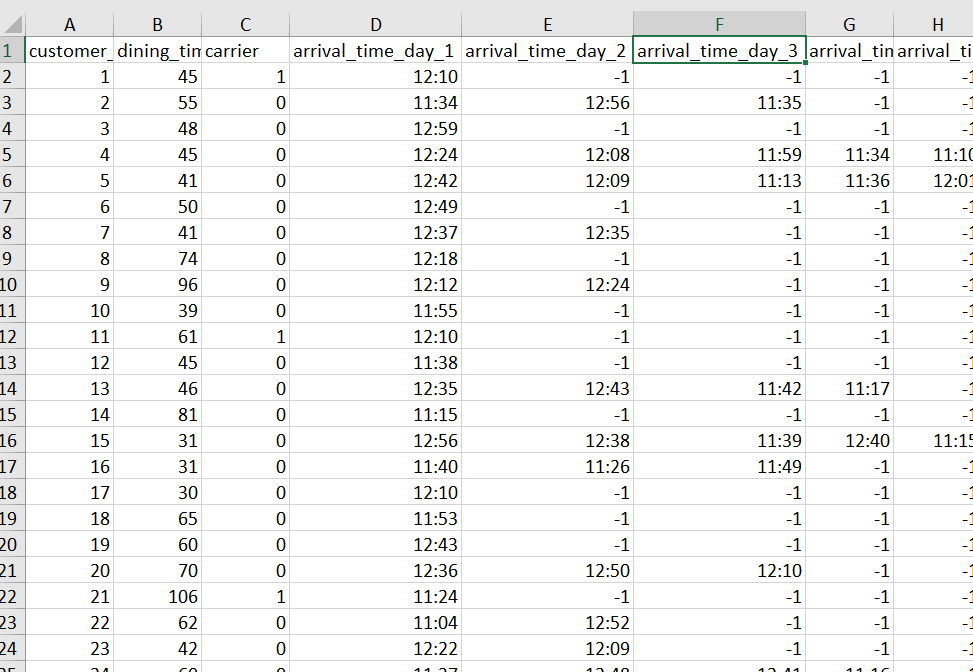
**There are still some printed data in the console for debugging, you can ignore them**

**Here are some output showing:**

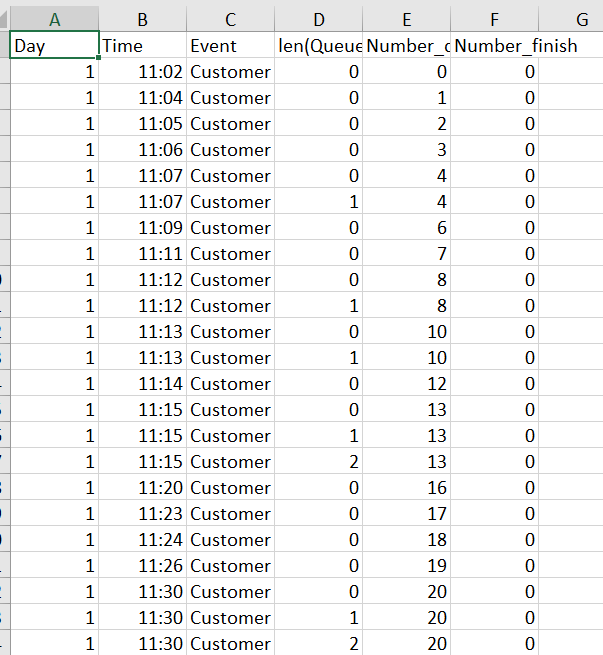
Output of config\_seed0.csv



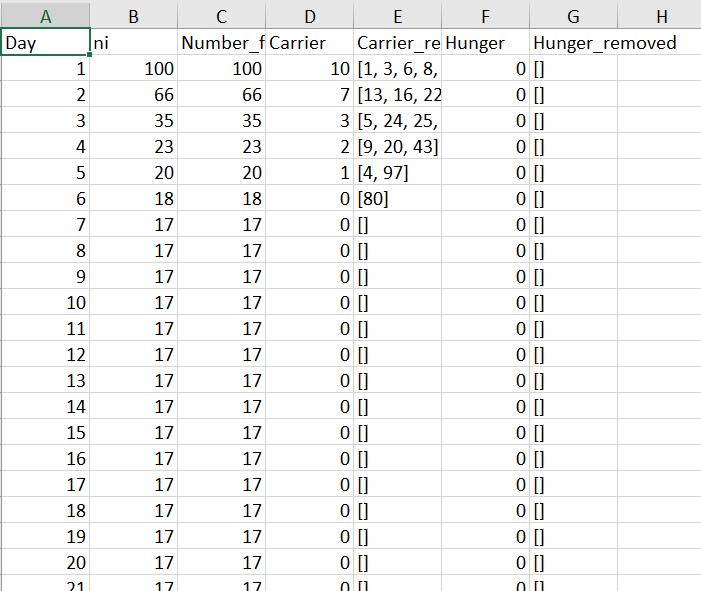
Output of customer\_seed0.csv



Output the result of the simulation: simulation\_result.csv or simulation\_result\_v1.csv



Output the summary of the simulation: simulation\_summary.csv or simulation\_summary\_v1.csv

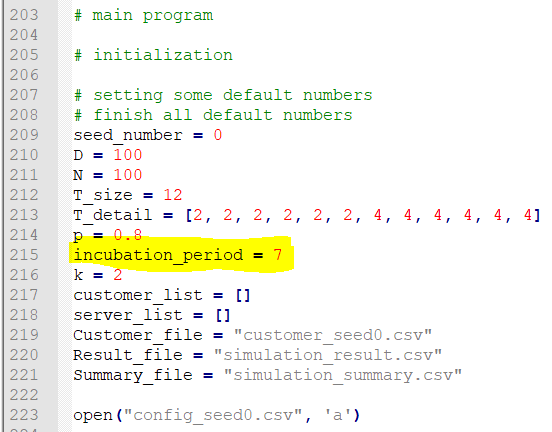


**Task 0:**

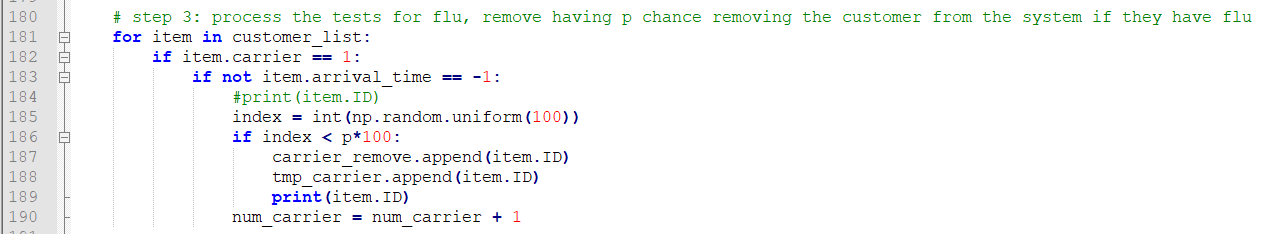
I will choose task 1, task 5, task 3.

**Task 1:**

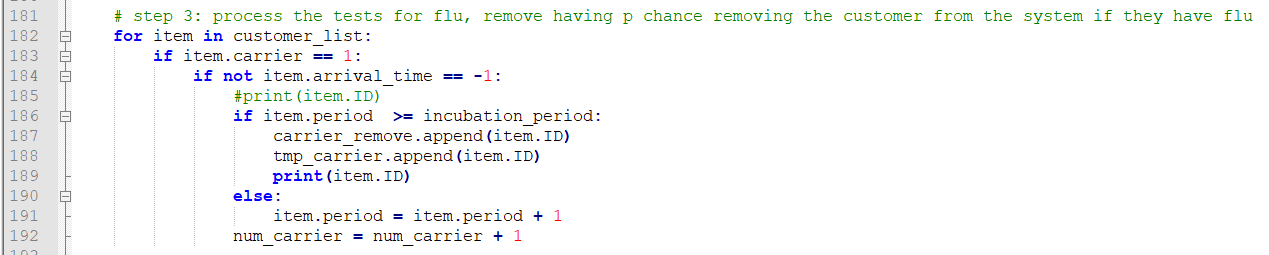
You can modify the period in this line in assignment4\_basic\_task1



Here is the old process for testing flu

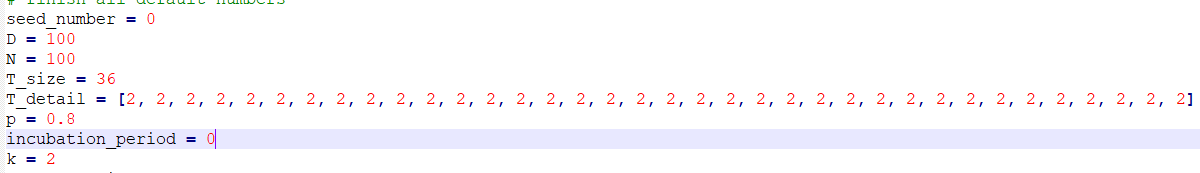


New program changed for task 1



**Experiment result:**

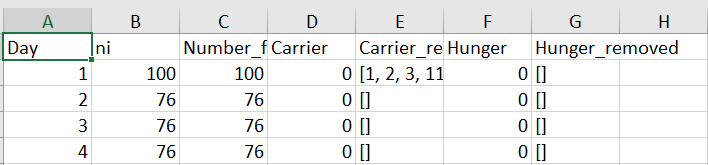
If we use the original data, the customer will get infected too fast, then we cant observe the relation clearly, so we limit the number of the customers in each deck, to reduce the rate of infection.



Only two customers can be in the one table

Also we add some more tables.

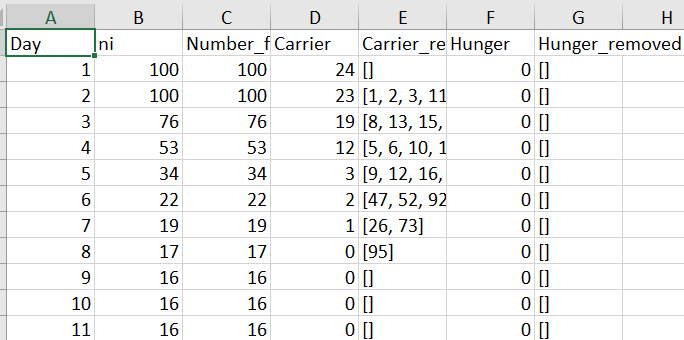
incubation\_period = 0

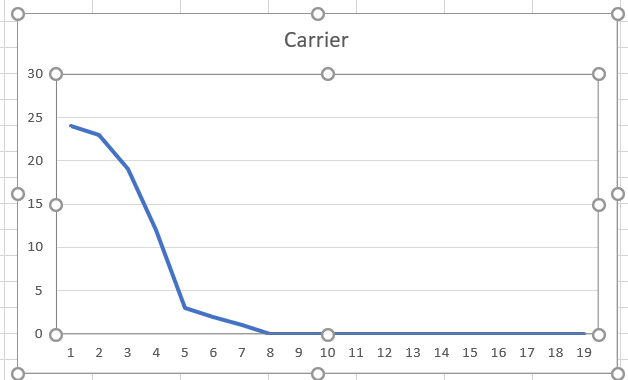


This is a control experiment, it shows day 1 will infect 24 people.

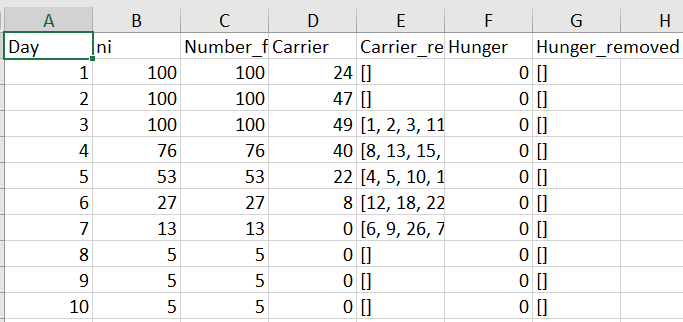
We can see the infect speed of this set up is slow, so we can observe the relation easily.

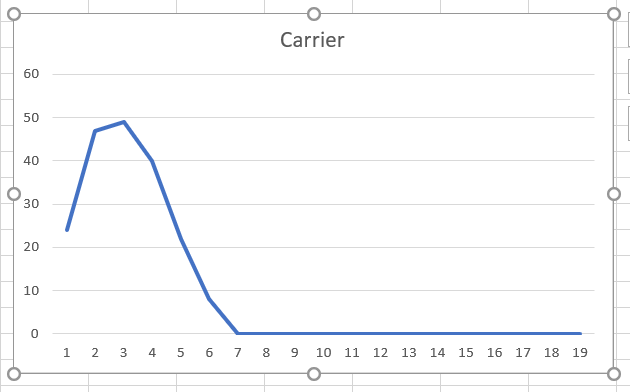
incubation\_period = 1



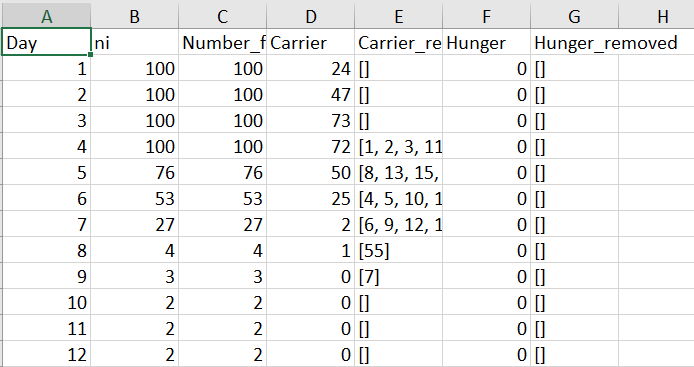


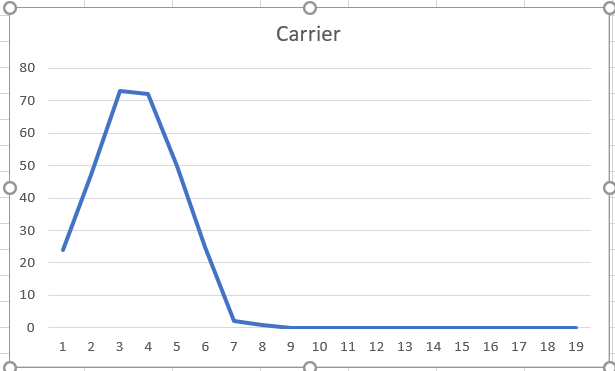
incubation\_period = 2



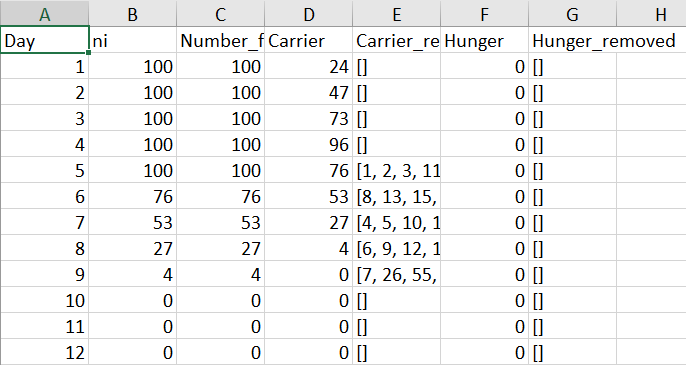


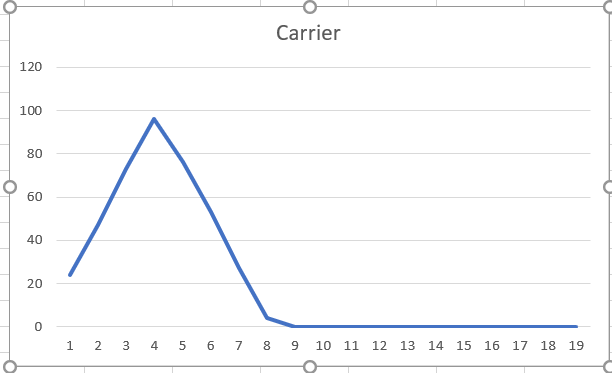
incubation\_period = 3



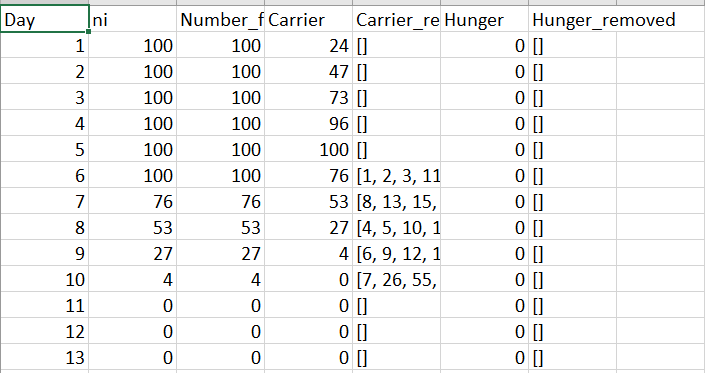


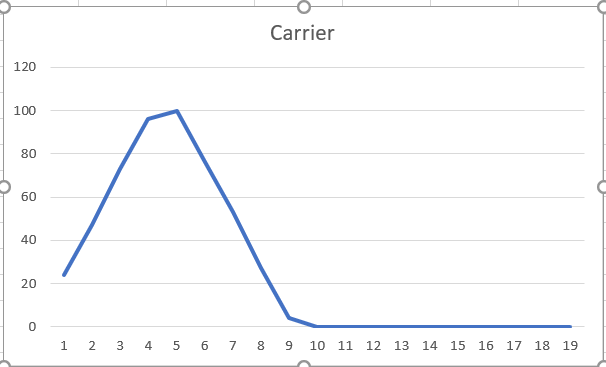
incubation\_period = 4





incubation\_period = 5





The graph are showing the number of carrier in the canteen on each day.

After testing incubation\_period 1-5, we can clearly see that with smaller incubation\_period, the spread of the flu will be smaller and the final number of infected customer is fewer.

We can see for incubation\_period 1-3,the final number of customer got infected (remain 16,5,2 respectively) decreases when incubation\_period increases, the peak of the number of carrier in the first 20 days (24,49,73 respectively) increase with the incubation\_period.

The slope of the graph shows the rate of the spread of the flu, the higher is the slope, the flu will spreads faster, as we know the more infected customer exists at the same time, the flu will spread much more faster.

So if the incubation\_period is smaller, then we can blacklist the infected customers earlier, then we can reduce the peak of the number of carrier and the slope of the graph earlier, before the peak got much higher.

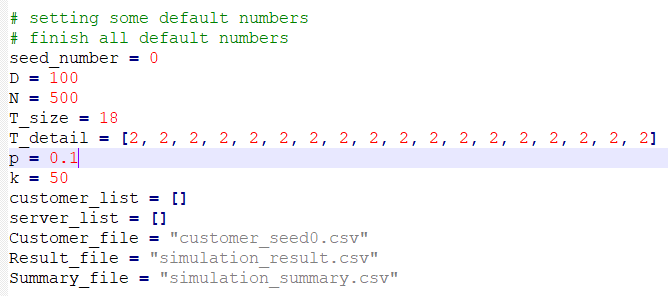
Therefore, we can conclude that the speed of the spread of the flu will increases with the length of the incubation period.

**Task 5:**

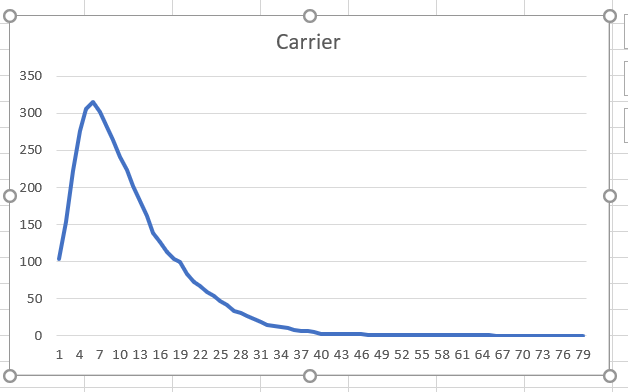
The steady state is the constant or steady state that is established after a certain time in the system or simulation. The transient state means the state between the beginning of the event and the steady state, the value in this state is still varying, and it will become steady and reach steady state after a certain time later.

**I really don’t understand what the question means “convert to different steady states after 50 days”, so I need to have steady state before or after day 50? Or I just need to make a graph with steady state? So I try to make some strange steady states which can fulfil all conditions.**

**Experiment one:**

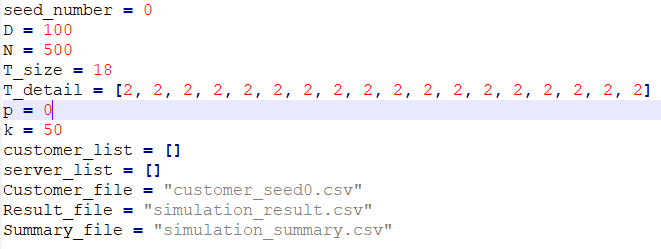
****

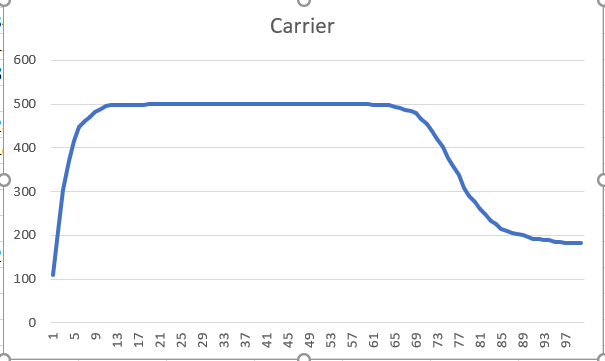
**Value of k can be ignore, as it cant affect the result here.**

****

We can see here, we have extreme low p, the sharp transient state is at around day 1 to day 30 as all customer got infected and then got blacklisted slowly after that, then finally there is no customer, it gradually convert to steady state after day 50.

**Experiment two:**





In this experiment, we would not blacklist customer, but we will have extreme large number of customers, and they can be hunger for 50 days! So we can make sure we will have two transient state and two steady state in this experiment.

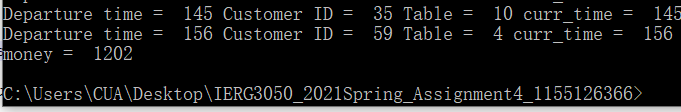
At day 1 to 10, customer starts to infect flu, this is a transient state, then at day 13 and day 65, most of the customers infect flu, then we have a first long steady state. Until around day 69, many hungry customers will be angry and will not come again, then the experiment will experience second huge change, then we have a transient state at day 69 to 80, then after many hungry customers leave, then we will convert to the second steady state after day 90. After that it should be steady forever (as this simulation is too long, I cant show it more after day 100).

In order attain steady state after day 50, I try to use some extreme input, so it can extent the process of transient state, and slow down the process to steady state.

In conclusion, in order to make transient state, we need to add some change to the system, then the system will process the change and finally reach the steady state, so I try to add some change after day 50 or some change can still affect the system after day 50, to make sure the steady states are located after day 50.

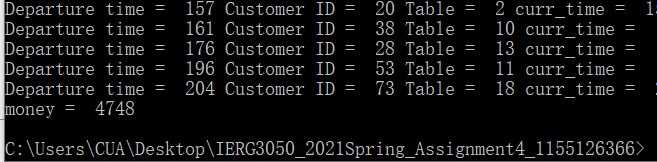
**Task 3:**

I try to make the program will count money, each customer will give 1 coin after dined.

****

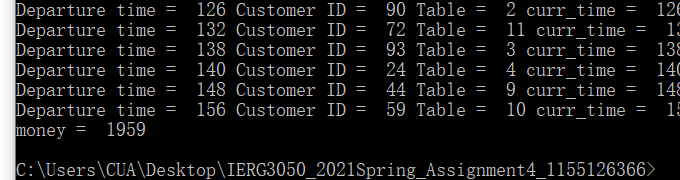
**T\_size = 18**

**T\_detail = [2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2]**

**For using all three 2-person tables**

**T\_size = 12**

**T\_detail = [4,4,4,4,4,4,4,4,4,4,4,4]**

**For using all Two 4-person tables**

**So I will use all three 2-person tables, we can make more money**