You have been hired by a grocery store that is considering switching over from a multiple line set up to a single line setup to handle customers when they check out. Before doing that, they have asked you to simulate if a single line can handle more customers than a multi-line format.

You are going to create these two simulations.

For **Part 1**, you will create a single-line simulation.

To help simulate this, we will use a time variable and a loop.

- New customer added every 5 ticks to a central queue
- cashier 1 - checks out a customer every 12 ticks
- cashier 2 - checks out a customer every 19 ticks
- cashier 3 - checks out a customer every 25 ticks

The starter code prompts the user for the number of customers. Time should start at 1 with 1 customer in the queue. At time = 5, you should add the second customer to the queue. At time = 10, the third customer is added. At time = 12, cashier 1 should remove the front of the queue. You will continue to add to the queue every 5 ticks and remove from the queue every 12, 19, and 25 ticks until all customers have been added to the queue and removed by a cashier.

Once all customers have been checked out, you should print out the total time needed.

** Partial Sample Output:**

```
How many customers do you want to simulate? 50

Cashier 1 served customer 1

Cashier 2 served customer 2

Cashier 1 now serving customer 3

Cashier 3 now serving customer 4

...

Cashier 1 served customer 48

Cashier 2 served customer 49

Cashier 1 served customer 50

It took a total of 289 ticks to serve all customers with a single line.
```

For **Part 2**, you will create a multi-line simulation.

We will use the same time setup and cashier processing times, but this time you will add customers to one of three cashier lines.

- New customer added every 5 ticks to the shortest queue (starting with cashier 1)
- cashier 1 - checks out a customer from their queue every 12 ticks
- cashier 2 - checks out a customer from their queue every 19 ticks
- cashier 3 - checks out a customer from their queue every 25 ticks

The starter code prompts the user for the number of customers. Time should start at 1 with 1 customer in the cashier 1 queue. At time = 5, you should add the second customer to the queue (cashier 2 at this point). At time = 10, the third customer is added to cashier 3's queue. At time = 12, cashier 1 should remove the front of their queue. At time = 15, the shortest queue is cashier 1, so you will add the new customer there. You will continue to add to the queues over time and check out customers as you get to them.

Remember, you should start looking for the shortest line with cashier 1. If all three have the same length, add to cashier 1's queue first. If cashier 2 and 3 have the same length, add to cashier 2's queue.

Once all customers have been checked out, you should print out the total time needed.

** Partial Sample Output:**

```
How many customers do you want to simulate? 50

Cashier 1 served customer 1

Cashier 2 served customer 2

Cashier 1 served customer 4

Cashier 3 served customer 3

...

Cashier 1 served customer 50

Cashier 2 served customer 48

Cashier 3 served customer 49

It took a total of 301 ticks to serve all customers with three separate lines.
```

Are there any conditions under which 3 separate lines takes less time than a single line? If so, what are they?

What suggestion(s) do you make to the grocery store that hired you?