Elijah Smith
COP4520 HW2
P2: Crystal Vase

The guests should choose the third strategy for forming a queue to enter the room. In the previous two cases, there is no guarantee that every guest will ever be able to enter the room if they arrive at a bad time to check and decide to leave to enjoy the party again later. Even if they did stay around the room, there's no guarantee they would be selected to enter if there are other guests waiting around too. Only the third strategy guarantees each guest to see the vase at least once. The downside to this approach is that guests are not able to leave and mingle before returning to check again, instead having to wait in the queue to enter the room. This is a fair tradeoff for being guaranteed admission.

In this implementation, each guest is represented by a GuestThread. Each GuestThread picks a random delay (upper-bounded by a constant) before the next time they enqueue to enter the room. When this time elapses, the GuestThread enqueues and waits to be notified if there are any GuestThreads enqueued in front of it. When notified, the thread will sleep for a random amount of time (also upper-bounded by a constant) that they decide to spend inside the showroom. On their way out, they remove themselves from the front of the queue and notify the next thread they can now wake and enter. The guest then decides randomly whether to enqueue another time (after some new random delay) or to finish execution.

The table below shows the average number of views per guest and simulation execution time based on several parameters:

Guest Count (N)	100	100	500	500	1000	1000
Max Enqueue Delay (ms)	10	200	10	5	10	10
Max Time in Room (ms)	5	100	5	10	15	15
Chance to run again (%)	75%	75%	50%	50%	25%	10%
Average Num of Views Each	4.14	3.43	2.00	1.96	1.33	1.12
Simulation Execution Time (ms)	1134	17253	2732	5051	10512	9068