Übung 3:

# Erweiterung der Restaurant Simulation

## Übernahme von Klassen ohne Änderungen

Folgende Klassen können übernommen werden:

* Buffet: Es gibt neue Buffetobjekte, diese bedürfen aber keiner Klassenänderung
* SimulationStatistics: Keine Änderungen nötig
* Log: Keine Änderungen nötig

## Übernahme von Klassen mit Änderungen

Folgende Klassen müssen angepasst werden:

* SimulationScheduler: Im Scheduler müssen sowohl die neuen Buffets, Queues, Cashiers und Gäste instantiert werden
* Counter: Die Klasse Counter wird langsam aber sicher trickreich, da eine Fallunterscheidung zwischen der Essensausgabe und der Klasse nötig wird:
  + Für die Essensausgabe an den Theken muss die Klasse so umgestaltet werden, dass bei hoher Auslastung eine weitere Person für die Essensausgabe abgestellt und später auch wieder abgezogen werden kann. Diese Vorgänge werden durch den Restaurantleiter ausgelöst und wirken sich auf die Abarbeitungsgeschwindigkeit der Queue, nicht aber deren Anzahl aus
  + Für die Bezahlung an der Kasse gilt, dass nach dem Öffnen einer Kasse immer nur eine Person abgestellt wird und bei Bedarf einfach eine weitere Kasse geöffnet wird. Der Restaurantleiter kann aber die Schliessung einer Kasse beantragen, was das Umverteilen der aktuellen Queue auf eine andere Kasse nötig macht
* Queue: Damit der Restaurantleiter die Länge der Schlange beobachten kann, muss er auf die Länge der Schlange in der Klasse Queue zugreifen können

Hinweis: Die Gäste müssen beim Bezahlen jetzt entscheiden, an welcher kürzesten Kasse sie sich anstellen möchten. Entweder kann dies der Gast selber bestimmen oder aber er stellt sich an einer Komposition aus Cashier-Queues und Cashier-Countern an. Diese Komposition leitet ihn dann an die beste Kasse weiter.

Wählt man den zweiten Ansatz, bereitet auch die Fallunterscheidung zwischen Essensausgabe und Kasse kein Problem, da beim Abstellen einer zweiten Person an der Essensausgabe einfach die Bedienzeit schneller wird. Für das Eröffnen und Schliessen einer Kasse mit Umverteilung ist die Komposition verantwortlich und nicht der Counter selber (Der Restaurantleiter steuert dann diese auch an). Deshalb:

* GuestA/GuestB: Die beiden Gästeklassen stellen sich jetzt an die Kassenkomposition an, welche sie intern an die beste Queue weiterleitet

## Neuentwicklung von Klassen

* GuestC: Gast C muss komplett neu entwickelt werden und das in der Aufgabenstellung beschriebene Verhalten umsetzen
* CashierComposition: Die Komposition für Cashier-Queues und Cashier-Counters muss auch neu entwickelt werden