Praktikum 3 - Protokoll

[Aufgabe 3.1](#h.2790k82lqkr7)

[Thread](#h.omxk89yypin9)

[Programm](#h.wl46j63kair7)

[Aufgabe 3.2](#h.qdqugtyhr3lh)

[Koch](#h.k5gh7ci7wegk)

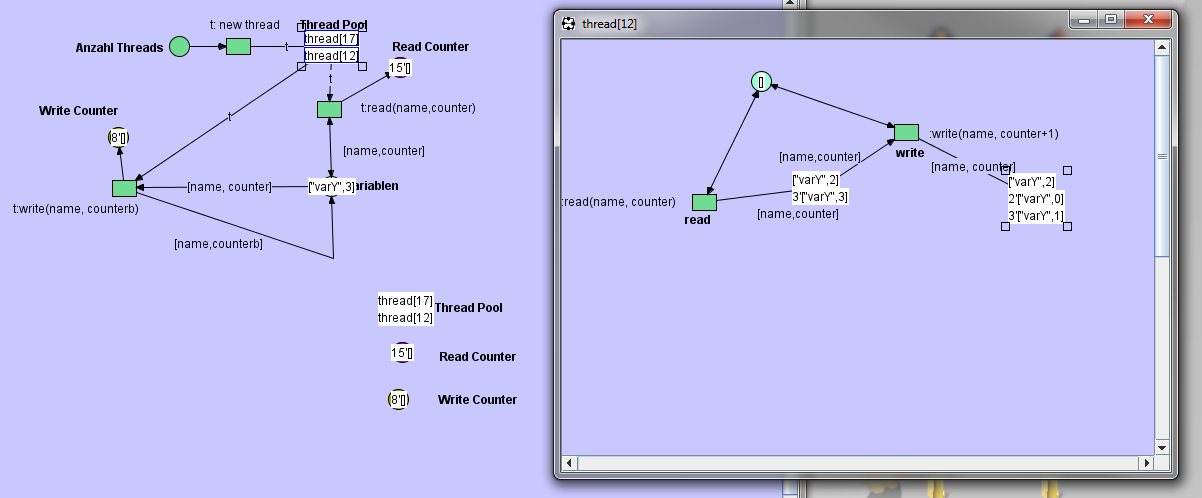
[Rezept](#h.ad200itcxkky)

[Arbeitsplatz](#h.8u4g88cumxbv)

[Küche](#h.y1n1jbs2pur)

# Aufgabe 3.1

## 



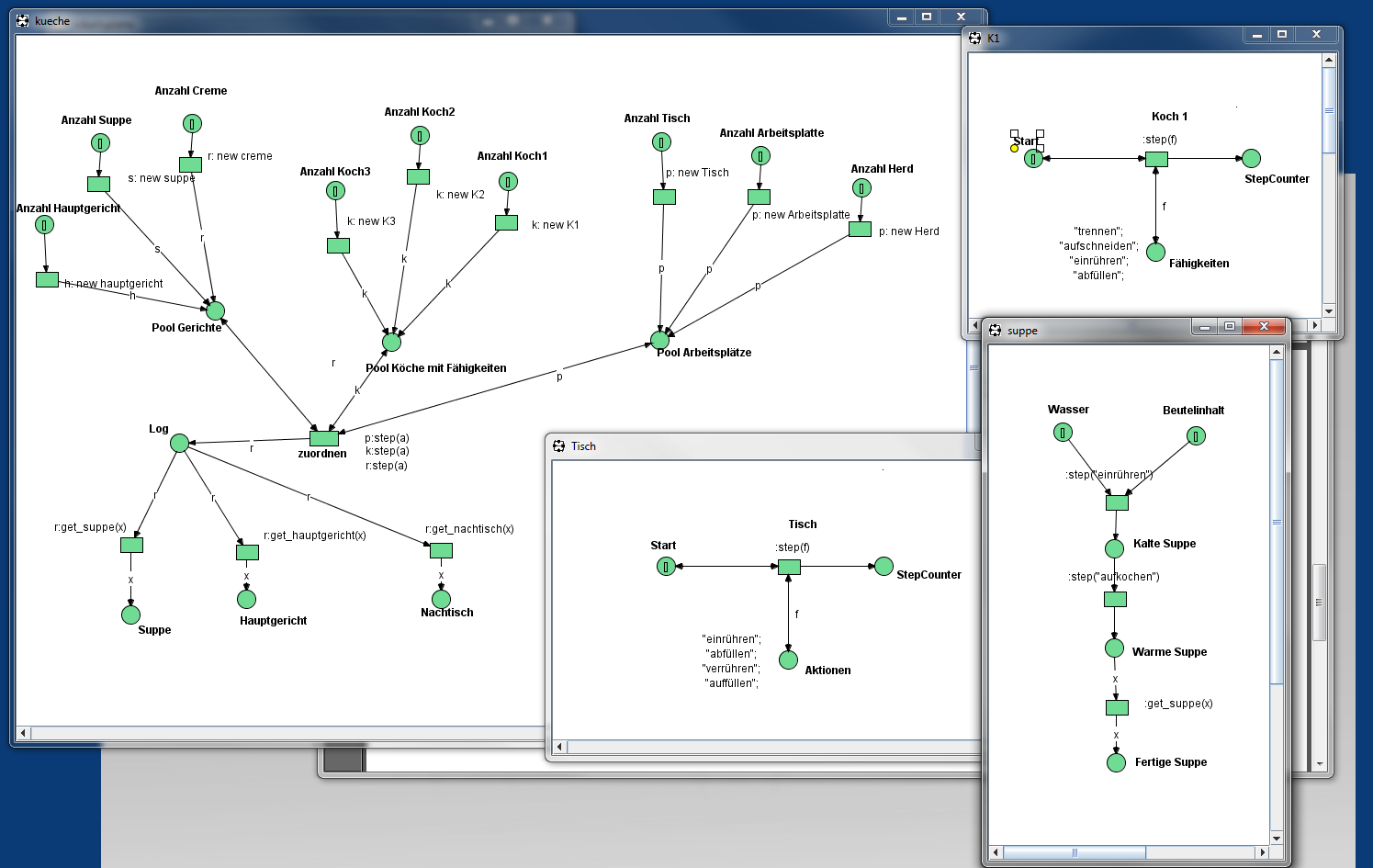
## Thread

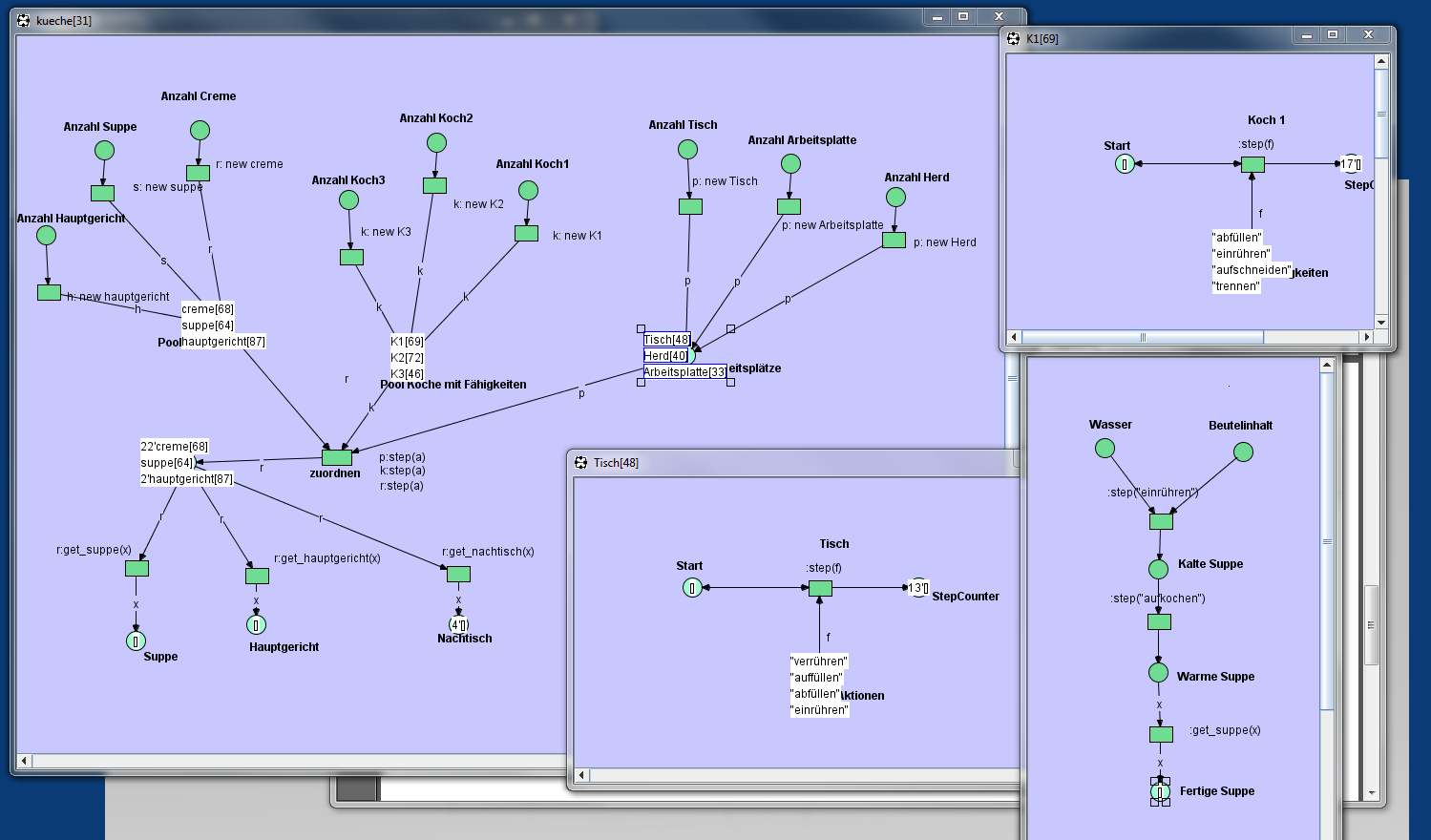
Wir erstellen 2 Threads mit je 3 Plätzen und 2 Transitionen - *read* und *write*. Diese können aus dem Programm aufgerufen werden, um die jeweilige Aktion auszuführen. Für die Methode *write* existiert eine Einschränkung. Damit sie aufgerufen werden kann, muss zuerst *read* aufgerufen werden. Also *read* von *write*.

## Programm

Im Programm erstellen wir ein Thread-Pool, Read- und Write-Counter. Die letzen sind dafür zuständig die Lese bzw. Schreibzugriffe zu zählen. Die Anzahl der Threads wird durch die Anzahl der Tokens auf dem Platz gesteuert. Die neuen Threads landen im ThreadPoll. Aus dem ThreadPoll werden die read und write Methdoen des Threads aufgerufen, die die Variablen aus dem Platz “Variablen” modifizieren.

# Aufgabe 3.2





## 

**Küche**

Wir modellieren in unserer Küche drei Rezepte (creme, suppe, hauptgerchit), 3 Köche mit unterschiedlichen Fähigkeiten und 3 verschiedene Arebitsplätze, bei denen unterschiedliche Aktionen möglich sind. Die Aktionen im Rezept, Fähigkeiten der Köche und Aktionen in den Arbeitsberiechen sollen jeweils einander zugeordnet werden (Transition *zuordnen*). Die fertigen Gerichte werden abschließend abgeholt.

## Koch

Jeder Koch hat eine Methode step der einen Arbeitschritt seiner Fähigkeit darstellt. Ein Counter zeigt wie oft ein Koch aktiv war.

## Rezept

Jedes Rezept wird schrittweise mit den Step-Methoden, die je unterschiedliche Aktionen erfordern, abgearbeitet. Eine zusätzliche get-Methode wurde angefügt um das fertige Gericht der Küche bereitzustellen.

## Arbeitsplatz

Jeder Arbeitsplatz hat eine Methode step der einen Arbeitschritt seiner möglichen Aktionen darstellt. Ein Counter zeigt wie oft an einem Arbeitsplatz gearbeitet wurde.

## 