

Ludwig-Maximilians-Universität München  
Prof. Dr. D. Kranzlmüller, Xiao-Ting Michelle To,  
Sophia Grundner-Culemann, Korbinian Staudacher

---

## Blockpraktikum Quantencomputing – Aufgabenblatt 4

---

### Challenge

Die folgende Aufgabe ist die eigentliche Challenge des Praktikums. Am Freitag, 14. Oktober, wird das gewinnende Team ermittelt; ausgewertet werden dabei die Lösungen, die um 15:00 Uhr im gitlab oder in Uni2Work verfügbar sind.<sup>1</sup> Sie dürfen danach bis Sonntag, 16. Oktober, 20:00 Uhr (also bis zum Abend vor der Prüfung) weiter optimieren und Lösungen im gitlab-Teamordner abgeben. (Abgabeformat .py oder .ipynb)

Es gewinnt das Team, welches zur Deadline für die schwerste Aufgabe, welche (bis dahin) überhaupt gelöst wurde, den Schaltkreis mit der kleinsten “Quantenfläche” einreicht. Als Quantenfläche bezeichnen wir dabei den Wert *Anzahl Qubits*  $\times$  *Quantenkosten*, wobei die Quantenkosten wie in Aufgabe 1.3 berechnet werden.

Das Lösen eines Logicals wurde im vorherigen Übungsblatt bereits besprochen. Zur Erinnerung: Logicals sind Rätsel, welche durch logisches Schlussfolgern gelöst werden. Dabei gibt es mehrere Kategorien mit jeweils gleich vielen Elementen und Hinweise welche der Elemente verbunden sind, bzw. welche nicht. Man kann solche Rätsel grafisch in einem Gitter lösen.

Lösen Sie nun die folgenden Logicals mit einem Quantencomputer. Sie haben auf dem Simulator 32 Qubits zur Verfügung und sollen den Algorithmus von Grover zur Lösung verwenden.

Die Schwierigkeitsgrad zum Finden einer Lösung für folgende Probleme mit einem Quantenalgorithmus steigt mit der Größe des Logicals. Insbesondere wächst mit steigendem Freiheitsgrad (also der Zahl unausgefüllter Felder) die Anzahl benötigter Qubits bzw. die Größe des Schaltkreis. Deshalb ist ein wesentlicher Aspekt der *Challenge*, den Schaltkreis für Grover so zu optimieren, dass Sie mit den zur Verfügung stehenden Qubits auskommen.

Sie müssen **nicht** zwingend alle Teilaufgaben bearbeiten. Schauen Sie einfach, wie weit Sie kommen.

Für die Prüfung ist die Platzierung bei der Challenge höchstens interessant, aber nicht ausschlaggebend.

---

<sup>1</sup>Stellen Sie sicher, dass Ihr Code an irgendeiner Stelle die Kosten der Lösung ausgibt.

#### Aufgabe 4.1: Logical mit 4 Kategorien und jeweils 3 Elementen (Challenge)

Nach dem Essen müssen einige aus dem QC-Team noch Dokumente für das Praktikum drucken. Sie schicken ihre Druckaufträge an verschiedene Drucker (namens Manet, Mergenthaler und Bullock) zu verschiedenen Zeiten. Finden Sie heraus, wer wann welches Dokument auf welchem Drucker druckt.

- Korbinian hat um 12:43 Uhr seinen Druckauftrag versendet. Das Dokument wird nicht mit Manet gedruckt.
- Das Dokument QCPrak\_01 wird um 12:36 Uhr gedruckt. Der Druckauftrag mit dem Namen „skript“ wurde an Bullock gesendet. *skript v QCPrak*
- Sophias Dokument hat einen Namen, der weniger als 10 Zeichen lang ist. Ein Druckauftrag wurde um 12:36 Uhr an Mergenthaler gesendet.

		01 10 11			01 10 11			01 10 11		
		Dokument			Zeit			Drucker		
		skript	QCPrak_01	meine_notizen	12:34	12:36	12:43	Manet	Mergenthaler	Bullock
Name	Sophia	1	X	1	1	1	1	1	X	1
	Korbinian	X	1	1	1	1	X	1	1	X
	Michelle	1	1	X	X	1	1	X	1	1
Ort	Manet	1	1	X	X	1	1			
	Mergenthaler	1	X	1	1	X	1			
	Bullock	X	1	1	1	1	X			
Zeit	12:34	1	1	X						
	12:36	1	X	1						
	12:43	X	1	1						

*Michelle - Zeit 12:36*  
*Michelle - Drucker*  
*Manet*

*11 - Sophia*

- Überlegen Sie sich erneut, wie Sie die Kategorien und Elemente in QuBits kodieren können. Eventuell reichen Ihnen nun die 32 QuBits nicht mehr aus. Was könnte man also tun, um sich QuBits zu sparen?
- Konstruieren Sie sich aus den gegebenen Hinweisen nun ein Orakel. Eventuell kann man auch Ancilla QuBits sparen, je nach Ihrer Kodierung.
- (**Challenge**) Versuchen Sie, die Anzahl der QuBits und/oder die Anzahl an QuBit-Operationen so gering wie möglich zu halten.

**Aufgabe 4.2:** Logical mit 4 Kategorien und jeweils 4 Elementen (optional)

**Achtung:** Diese Aufgabe ist *optional*. Diese ist für den Fall, dass Sie früher fertig sind, keine Optimierungsidee mehr haben und sich noch an einem schwierigeren Logical versuchen wollen. Sie können auch die Lösung dieses Logical in der Prüfung präsentieren.

Nach dem anstrengenden Tag geht ein Teil des Teams noch zusammen ein Eis essen und geht danach nach Hause. Finden Sie heraus, wer welches Eis mit welchem Topping hatte und wie die einzelnen Personen nach Hause fahren.

- Michelle isst Cheesecake-Eis. Andreas fährt nicht mit dem Bus nach Hause.
- Sophia nimmt die Tram. Sie hat kein Topping mit Nüssen.
- Das Dunkle-Schokolade-Eis hat keine Sahne.
- Die Person, die mit dem Fahrrad nach Hause fährt, isst das Mango-Lassi-Eis. Die Person, die die Bahn nimmt, hat keine Nüsse.
- Das Schmand-Eis hat als Topping Schokoladenstreusel. Es wurde nicht von Sophia bestellt, und die Person, die das Eis bestellt hat, fährt nicht mit der Bahn nach Hause.

		Eissorte				Topping				Fahrt			
		Schmand	Cheesecake	Mango Lassi	Dunkle Schokolade	Nüsse	Erdbeeren	Schokoladenstreusel	Sahne	Tram	Bus	Bahn	Fahrrad
Name	Andreas												
	Sophia												
	Korbinian												
	Michelle												
Fahrt	Tram												
	Bus												
	Bahn												
	Fahrrad												
Topping	Nüsse												
	Erdbeeren												
	Schokoladenstreusel												
	Sahne												

- Überlegen Sie sich erneut, wie Sie die Kategorien und Elemente in QuBits kodieren können. Eventuell reichen Ihnen nun die 32 QuBits nicht mehr aus. Was könnte man also tun, um sich QuBits zu sparen?
- Konstruieren Sie sich aus den gegebenen Hinweisen nun ein Orakel. Eventuell kann man auch Ancilla QuBits sparen, je nach Ihrer Kodierung.
- (Challenge)** Versuchen Sie, die Anzahl der QuBits und/oder die Anzahl der QuBit-Operationen so gering wie möglich zu halten.