

# Den Grover-Algorithmus verstehen

## Arbeitsblatt mit Lösungen

Ziel ist es, den Grundgedanken und die Funktionsweise des Grover-Algorithmus zu erfassen, insbesondere wie er die Suche in einer unsortierten Liste von Elementen beschleunigen kann. Für diese Aufgabe müsst ihr zu fünft sein.

### Schritt 1: Vorbereitung

- Erstellt eine Liste mit vier Elementen (dies können Zahlen, Buchstaben oder Objekte sein, z. B. A, B, C und D).
- Wählt eines der Elemente aus. Es ist das „versteckte Element“, das gefunden werden soll, beispielsweise das Element C.

### Schritt 2: Einen klassischen Suchvorgang simulieren

#### 1. Rollenspiel

Eine\*r von euch spielt eine Person, die auf einem klassischen Computer nach einem versteckten Element sucht. Die anderen vier sind die vier Elemente A, B, C, D. Die vier Schüler\*innen wissen, wer von ihnen das versteckte Element ist, die suchende Person weiß es nicht.

#### 2. Abfrage

Die suchende Person zeigt jeweils auf eine\*n der Mitschüler\*innen und fragt: „Bist du das richtige Element?“ Die Schüler\*innen antworten mit „Ja“ oder „Nein“.

#### 3. Notiert die Anzahl der Abfragen

Schreibt euch auf, wie oft die suchende Person fragen muss, bis sie die Antwort „Ja“ bekommt.

#### 4. Diskutiert die drei Fälle für $N = 4$ .

| Fall                      | Was passiert?                                                                                                                                                                                                             | Anzahl der erforderlichen Abfragen |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| <b>Bester</b>             | Die erste Abfrage ist die richtige.                                                                                                                                                                                       | <b>1</b>                           |
| <b>Schlechtester</b>      | Das richtige Element ist das letzte, das überprüft wird.                                                                                                                                                                  | <b>4</b>                           |
| <b>Durchschnittlicher</b> | Jede der vier Positionen ist gleich wahrscheinlich, sodass die durchschnittliche Anzahl der Abfragen ungefähr $N/2 = 4/2 = 2$ beträgt.<br><br>Im Durchschnitt muss die suchende Person ihre Frage <b>zweimal</b> stellen. | <b>Im Schnitt 2</b>                |



## Schritt 5: Diskussion & Fragen

- **Wie unterscheidet sich das Orakel von einem klassischen „Hinweis“?**  
Es überprüft alle Gegenstände auf einmal (ohne die Gegenstände einzeln „abzufragen“) und markiert das versteckte Element, erhält jedoch keine Antwort wie „Dies ist (nicht) das versteckte Element“.
- **Was bewirkt der Verstärkungsschritt?**  
Der Verstärkungsschritt verringert die Wahrscheinlichkeit der nicht markierten Elemente und erhöht die Wahrscheinlichkeit, dass das markierte Element die richtige Antwort ist. Das versteckte Element sticht als wahrscheinliche Antwort schneller hervor als bei der Verwendung sequenzieller Abfragen in einer klassischen Suche.
- **Beschreibt den günstigsten, den schlechtesten und den durchschnittlichen Fall bei der Verwendung einer klassischen Suche, um einen Gegenstand unter 12 unsortierten Gegenständen zu finden.**

| Fall               | Was passiert?                                                                                                                                                                       | Anzahl der erforderlichen Abfragen |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------|
| Bester             | Das erste Element ist das gesuchte.                                                                                                                                                 | 1                                  |
| Schlechtester      | Man muss sich jedes einzelne Element anschauen, denn das gesuchte Element ist das letzte.                                                                                           | 12                                 |
| Durchschnittlicher | Das gesuchte Element kann sich mit gleicher Wahrscheinlichkeit an jeder Position befinden, sodass man sich im Schnitt die Hälfte der Elemente anschauen muss. Das sind $12/2 = 6$ . | Im Schnitt 6                       |

Ein klassischer Suchvorgang kann also sehr schnell sein, wenn ihr nur einmal fragen müsst, weil das versteckte Element an erster Stelle der Liste steht. Es kann aber auch sein, dass ihr 12 Mal fragen müsst (tatsächlich reichen 11 Mal, denn sobald ihr das Ergebnis der 11. Abfrage habt, wisst ihr, ob sich das Element an Position 11 oder 12 befindet), was eine Weile dauern kann. Im Schnitt sind 6 Abfragen erforderlich, um das versteckte Element zu finden.

1. **Überprüft, ob die durchschnittliche Anzahl der Abfragen 6 ( $= N/2$ ) beträgt, mit der im obigen Kasten beschriebenen Methode.**

Die möglichen Fälle sind:

**Fall 1 – ja nein nein nein nein nein nein nein nein nein nein** → erfordert 1 Abfrage = die erste Schülerin ist das gesuchte Element (**ja**), die anderen 11 nicht (**nein**).

**Fall 2 – nein ja nein nein nein nein nein nein nein nein nein** → erfordert 2 Abfragen.

**Fall 3 – nein nein ja nein nein nein nein nein nein nein nein** → erfordert 3 Abfragen.

**Fall 4 – nein nein nein ja nein nein nein nein nein nein nein** → erfordert 4 Abfragen.

**Fall 5 – nein nein nein nein ja nein nein nein nein nein nein** → erfordert 5 Abfragen.

**Fall 6 – nein nein nein nein nein ja nein nein nein nein nein** → erfordert 6 Abfragen.

**Fall 7** – nein nein nein nein nein nein ja nein nein nein nein nein → erfordert 7 Abfragen.  
**Fall 8** – nein nein nein nein nein nein nein ja nein nein nein nein → erfordert 8 Abfragen.  
**Fall 9** – nein nein nein nein nein nein nein nein ja nein nein nein → erfordert 9 Abfragen.  
**Fall 10** – nein nein nein nein nein nein nein nein nein ja nein nein → erfordert 10 Abfragen.  
**Fall 11** – nein nein nein nein nein nein nein nein nein nein ja nein → erfordert 11 Abfragen.  
**Fall 12** – nein nein nein nein nein nein nein nein nein nein nein ja → es ist keine zusätzliche Abfrage nötig: Wenn man weiß, dass das 11. Element **nein** ist, dann weiß man, dass das 12. **ja** ist.

Um die durchschnittliche Anzahl der Abfragen zu berechnen, muss man alle erforderlichen Abfragen addieren und durch 11 teilen (d. h. die Anzahl der möglichen Fälle; Fall 12 kann verworfen werden, da er redundant zu Fall 11 ist):

$$(1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 + 11)/11 = 66/11 = 6;$$

Die durchschnittliche Anzahl der erforderlichen Abfragen ist damit:  **$N/2 = 12/2 = 6$ .**

### 1. Wie viele Schritte wären erforderlich, um den Gegenstand mit dem Grover-Algorithmus zu finden?

Es ist sehr wahrscheinlich, dass ihr nach  $\sqrt{12} \approx 3,5$  Abfragen das richtige Element findet – also nach 3 bis 4 „Orakel fragen + Verstärkung“-Zyklen.