Worksheet
Understanding Grover's algorithm

Objective: To get a basic understanding of how Grover's algorithm works and how it can speed up a search in an unsorted list of items. You need to be a group of five students to carry out this exercise.

Step 1: Setting up the search problem

- Create a list of four items (these can be numbers, letters or objects, e. g. A, B, C and D).
- Choose one of the items as the hidden item to be found, for instance item C.

Step 2: Simulating a classical search

- Have one of you act as a user of a classical computer searching for the hidden item. The other four are assigned to the four items (without the user knowing who represents which item).
- The classical computer can only check one item at a time. The user is entering a query – i.e. asking one of the four students: "Are you the correct item?"
- The classical computer returns either "Yes" or "No". Track how many queries it takes before you find the correct item.
- Explain the **best case** (the item is found on the first guess), the **average case** (two guesses are needed) and the **worst case** (three guesses are needed because the correct item is the last one in the list).

Hint: for N items, the average number of queries with a classical search is $N/2$. In our case, $N = 4$, meaning that the average number of queries will be $4/2 = 2$.

> **How many queries are needed to find a hidden item in an unsorted list with four elements ($N = 4$)?**
>
> The possible cases are:
> **Case 1 – yes no no no** → requires 1 query = the first student is the hidden item (**yes**), the three students standing next to him/her are not (**no**).
> **Case 2 – no yes no no** → requires 2 queries.
> **Case 3 – no no yes no** → requires 3 queries.
> **Case 4 – no no no yes** → no additional query is needed, because if you know that the third item is **no**, then you know that the fourth item is **yes.**
>
> Since all the positions are equally probable, the average number of required queries is $\mathbf{N/2 = 2}$.

Step 3: Simulating a search with Grover's algorithm

1. Four students are assigned to the four items A, B, C, D, and one of them actually is the hidden item. Draw two lines on the floor, one is denoted "Yes" and the other one "No". The four students stand anywhere in between the two lines. They all have an equal probability to be the correct answer (= the hidden item).
2. The user-student repeatedly asks the oracle:[1]
"Where is the hidden item?". Each time, the student which is representing the hidden item takes a small step towards the "Yes"-line (but only very slightly, so that the user cannot be quite sure yet that he/she is the correct item). He/she is "marked". At the same time, the other three students take a small step towards the "No"-line. This is probably not easy to

---

[1] Asking an oracle is a method used in computing. It can, for example, solve a decision question (i.e. "this one or that one?"). The name refers to an oracle in Greek antiquity – a person or a thing that gave answers to questions about the future or offered guidance for a decision to be taken.

TAKEN FROM QUANTUM COMPUTING IN STEM EDUCATION
A project by Science on Stage Germany

grasp, but that is how the Grover algorithm works – it marks the object without specifically examining it.

3. Further amplification:
   If you repeat this step several times, the student representing the hidden item will move towards the "Yes"-line more and more while the others move towards the "No"-line. This increases the probability of the marked student, respectively decreases the probability of the other students to be the correct answer. Each iteration makes the "hidden" student stand out more clearly.

4. Measurement:
   After just two of these "mark and amplify" cycles, you will probably see which student is the "marked" item since he/she will be the one closest to the "Yes"-line. That student is the final "measured answer".

Step 4: Comparison with the classical search

Taking only four items doesn't show the real advantage over a classical search, it merely shows the principle of how the Grover algorithm works. The Grover algorithm limits the number of queries to find a hidden item among N items to approximately $\sqrt{N}$.
The benefit of using the Grover algorithm becomes clear very quickly when increasing the number of items. That is why you might want to repeat this exercise with a larger group of students, let's say 16. In this case, a classical search would need on average 16/2 = 8 queries, whereas Grover's algorithm typically requires on the order of $\sqrt{16} = 4$ queries.

Step 5: Discussion & questions

- How does the oracle differ from a classical "hint"?
- What does the amplification step do?
- Describe the best, worst and average case when using a classical search to find an item among 12 unsorted items.
- Verify whether the average number of queries is 6 (= $N/2$) with the method described in the box above.
- How many steps would be required to find the item using Grover's algorithm?