

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
“БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ”

Кафедра ИИТ

ОТЧЁТ
По лабораторной работе №5
«Алгоритм Гровера»

Выполнил:
Студент группы ИИ-22
Дубина Н.С.
Проверила:
Хацкевич А.С.

Брест 2024

Цель работы: ознакомление с алгоритмом Гровера и его реализацией в квантовых системах.

Задачи:

Найдите количество итераций с наибольшей амплитудой при запуске алгоритма Гровера с одним решением в базе данных с $N=2^7$. Как показано выше, измените количество итераций и проверьте усиление. Ответ должен быть целым числом.

Ход работы

```
from qiskit import QuantumCircuit
from qiskit_aer import Aer

import matplotlib

matplotlib.use('QtAgg')
import matplotlib.pyplot as plt

target_state = '1110111'
num_qubits = 7

def apply_oracle(qc):
    """Функция для применения оракула, помечающего целевое состояние."""
    for idx, bit in enumerate(target_state):
        if bit == '0':
            qc.x(idx)

    qc.h(num_qubits - 1)
    qc.mcx(list(range(num_qubits - 1)), num_qubits - 1)
    qc.h(num_qubits - 1)

    for idx, bit in enumerate(target_state):
        if bit == '0':
            qc.x(idx)

    qc.barrier()

def apply_diffuser(qc):
    """Функция для усиления амплитуды целевого состояния."""
    qc.h(range(num_qubits))
    qc.x(range(num_qubits))
    qc.h(num_qubits - 1)
    qc.mcx(list(range(num_qubits - 1)), num_qubits - 1)
    qc.h(num_qubits - 1)
    qc.x(range(num_qubits))
    qc.h(range(num_qubits))

    qc.barrier()

def run_grover_simulation(show_debug=False):
    """Запускает алгоритм Гровера для поиска целевого состояния с визуализацией."""
    backend_simulator = Aer.get_backend('qasm_simulator')
    success_counts = []
    max_iterations = range(15)

    for i in max_iterations:
        quantum_circuit = QuantumCircuit(num_qubits)
        quantum_circuit.h(range(num_qubits))
        quantum_circuit.barrier()

        # Применяем оракул и усиление амплитуды необходимое число раз
        for _ in range(i):
            apply_oracle(quantum_circuit)
            apply_diffuser(quantum_circuit)

        quantum_circuit.measure_all()
        results = backend_simulator.run(quantum_circuit).result()
        counts = results.get_counts()
        success_counts.append(counts.get(target_state[::-1], 0))
```

```

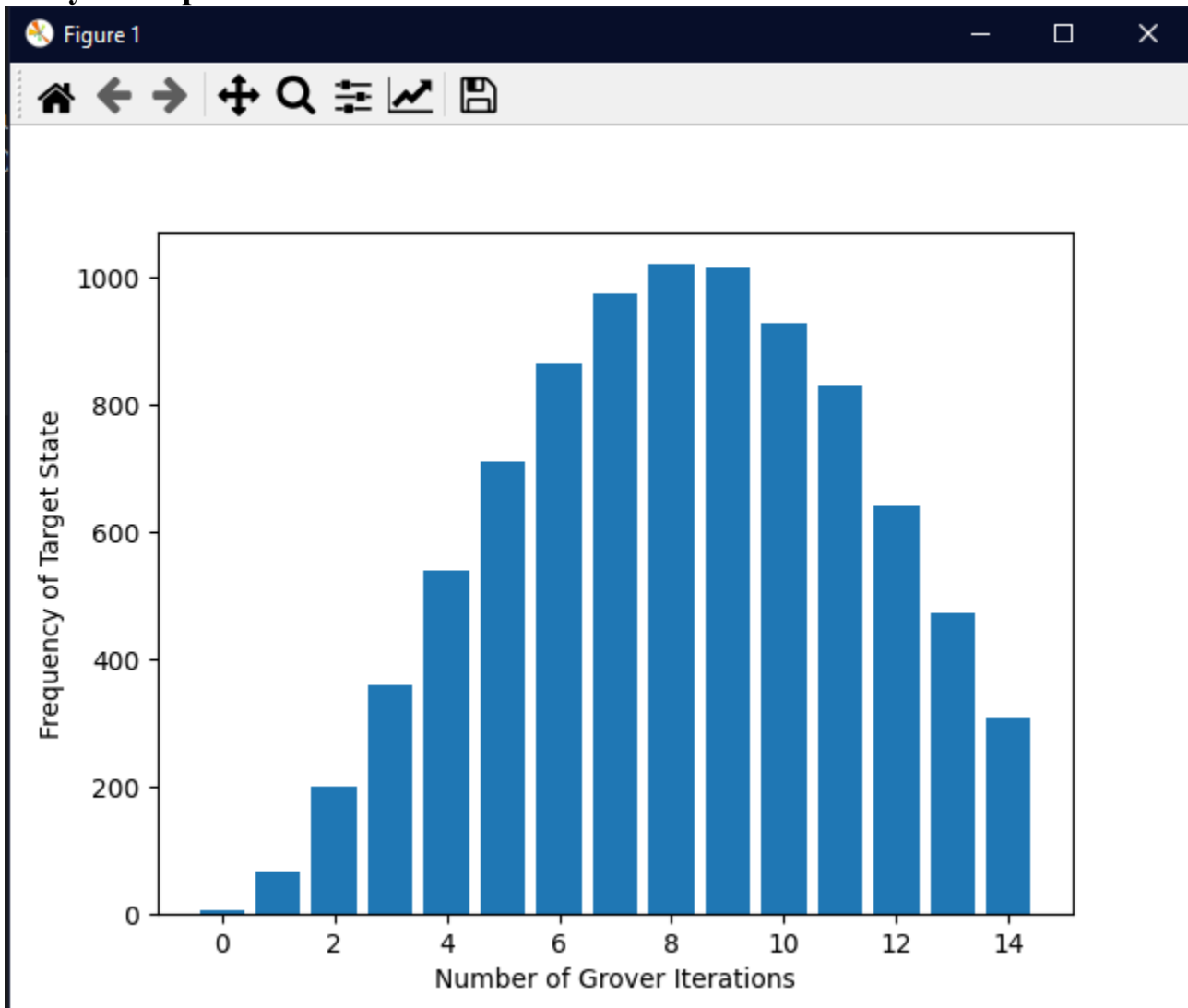
        if show_debug and i == 1:
            quantum_circuit.draw('mpl')
            plt.show()

    plt.bar(max_iterations, success_counts)
    plt.xlabel("Number of Grover Iterations")
    plt.ylabel("Frequency of Target State")
    plt.show()

if __name__ == "__main__":
    run_grover_simulation(show_debug=True)

```

Результат работы:



Вывод: ознакомился с алгоритмом Гровера и его реализацией в квантовых системах.