

TASK 7: Grover's Oracle (2-qubit)

Aim:

To construct and test a 2-qubit Grover oracle that marks a specified state.

Algorithm:

- Create a 4x4 identity matrix.
- Flip the sign of the target state's amplitude.
- Apply oracle to a uniform superposition.
- Observe phase inversion effect.

Program:

```
print("\n" + "="*50)
print("TASK 7: GROVER'S ORACLE (2-QUBIT)")
print("="*50)

def grover_oracle_2qubit(marked_state=3):
    """Create 2-qubit Grover oracle that marks specified state"""
    oracle = np.eye(4)
    oracle[marked_state, marked_state] = -1 # Mark the target state
    return oracle

# Create oracle for different marked states
oracle_0 = grover_oracle_2qubit(0) # Marks |00>
oracle_3 = grover_oracle_2qubit(3) # Marks |11>

print("Oracle marking |00>:")
print(oracle_0)
print("\nOracle marking |11>:")
```

```
print(oracle_3)
```

```
# Test oracle on uniform superposition
uniform_superposition = np.ones(4) / 2 # (|00> + |01> + |10>
+ |11>)/2
```

```
print(f"\nUniform superposition: {uniform_superposition}")
print(f"After oracle (|11> marked): {oracle_3 @
uniform_superposition}")
```

Result:

The 2-qubit Grover oracle was implemented and successfully marked the target state.