Stage 1: Implementing a QRNG on IBM QPUs

Objective:

The goal of Stage 1 was to implement a quantum random number generator (QRNG) on an IBM quantum machine or simulator to generate quantum random number (QRNG) data. By doing this, I explored the fundamentals of quantum circuits, qubit superposition, and measurement—building a strong foundation for understanding how quantum randomness is generated.

Description:

To meet this objective, I first used IBM Qiskit simulators to design a quantum circuit. The circuit employed Hadamard gates to place each qubit into a superposition state, which gives equal probability for the qubits to collapse to either 0 or 1 upon measurement. This random collapse is what forms the basis of quantum random number generation (QRNG).

Once I had a solid grasp of creating and measuring qubits in superposition, I moved to running the circuits on actual IBM quantum computers using the Qiskit platform. By connecting to IBM Quantum's API, I was able to execute the circuits on quantum devices like 'ibm_brisbane' and successfully retrieve real QRNG data.

What I Learned:

1. **Quantum Superposition:** Through this process, I learned that by applying Hadamard gates, qubits can be placed in a state where they have an equal chance of collapsing to 0 or 1. This principle underpins the generation of quantum randomness.

2. **Qiskit Basics**: I became proficient with Qiskit, learning how to build quantum circuits, simulate them locally, and execute jobs on both simulators and real quantum hardware. I also explored visualizing and interpreting the results.

3. **Job Execution on IBM QPUs:** I gained practical experience in submitting and managing quantum jobs through the QiskitRuntimeService, learning how to utilize IBM Quantum's API and working with real quantum devices.

Coding Process Overview:

- **Circuit Creation:** I developed a function to create a quantum circuit using Hadamard gates to place qubits into superposition. This setup is essential for generating the random measurements that form the basis of the QRNG.

- **Simulating the Circuit:** I used the Qiskit AerSimulator to locally test the quantum circuit by running it multiple times and capturing the resulting bitstrings.

- **Running on Real QPUs:** After confirming that the simulator worked as expected, I submitted jobs to real IBM quantum devices like 'ibm_brisbane'. This allowed me to generate genuine QRNG data from physical quantum machines.

- **Combining with PRNG Data:** For comparison, I generated pseudorandom numbers (PRNG) and combined them with QRNG data. This allowed me to assess the differences between quantum and classical randomness.

- **Data Output:** Finally, I wrote the combined QRNG and PRNG data to a text file and visualized the distribution of 0s and 1s using histograms to better understand the randomness of each dataset.

Purpose:

This stage played a crucial role in deepening my understanding of quantum randomness, as well as how quantum circuits work in practice. The data generated from this stage became an important component in my subsequent work, including using QRNG for machine learning purposes later in the challenge. Furthermore, the ability to run circuits on real quantum hardware through IBM QPUs was invaluable in preparing for more advanced tasks in later stages.