Qiskit
Fall Fest 2024

Hello Student Leaders,

As an organizer of Qiskit Fall Fest 2024, we wanted to provide you with as many resources as possible!

In this document, we provide some hackathon ideas for you -- please feel free to choose any of these or, make your own unique hackathon problems! We have also provided sample judging criteria to help evaluate your participant submissions.

Intermediate Level:

- Quantum Circuit Transpilation Prompt

Advanced Level:

- Dynamic Circuit Prompt
- Go into The Real-World Prompt

Proposed Judging Criteria

This is a suggested rubric your judges can use, no matter what type of hackathon you decide to run. This is, of course, only a suggestion. Please modify or change where you see fit.

1. **Technical Aspects (30 total points)**

How complex is the quantum algorithm? Is it well optimized? Can the architecture serve users at a reasonable scale? How accessible is the end user application? Is it easy to use and intuitive for end users? Did the team use any significant parts of the Qiskit SDK, Qiskit Runtime, or other parts of the Qiskit ecosystem?

2. **Originality and Uniqueness (25 total points)**

How unique is this project compared to others? How interesting is it? Did the team attempt something new or difficult?

3. **Usefulness and Complexity (25 total points)**

How useful is the project and how well-designed is it? How functional is it at the time of judging? Can it be used in real-world business applications or serve as a valuable tool for individuals? Are there ways this project could be further built out and refined upon?

4. **Presentation (20 total points)**

How well did the team present their project? Were they able to explain their decisions? Did the entire team have a chance to speak? Did they tell a cohesive story?

Hackathon Prompt: Explore Quantum Circuit Transpilation

Background and motivation:

Transpilation of a quantum circuit is a complex process of converting the given circuit into the executable set of gates, matching with the topology of the target quantum device, using an optimal number of gates, and applying techniques to deal with errors. Depending on the performance of the transpiler, a circuit can shrink into one-fifth the size of the original circuit and achieve a significant reduction of errors physically. In this topic, you will explore the detailed steps of transpilation and play with your own or ready-made functions to optimize the given quantum circuit. The target circuit of this topic will be the Quantum Fourier Transform (QFT) circuit, which is one of the most important quantum subroutines, providing exponential speedup compared to classical computation.

Transpile the QFT circuit to get the best score. The circuit must perform the QFT with arbitrary inputs, matching with the target quantum device topology within the basic gate set. The score will be given based on the expected performance of the circuit, considering accuracy of each gate in the circuit.

You can find the grader file at here and the main repository of 2024 Quantum Korea Hackathon can be found here.

Getting started:

Qiskit 1.0 transpiler provides a powerful flexibility to users with its abundant options. Also, to facilitate the development and reuse of custom transpilation code by the wider community of Qiskit users, the Qiskit SDK supports a plugin interface that enables third-party Python packages to declare that they provide extended transpilation functionality accessible via Qiskit.

How about making your own transpiler and becoming a contributor? Creating a transpiler plugin is a great way to share your transpilation code with the wider Qiskit community, allowing other users to benefit from the functionality you've developed. Here you can find guidelines and instructions on how to contribute to the Qiskit community by contributing quality transpiler plugins.

Suggested resources:

- Qiskit Fall Fest 2024 Notebook 3
- Introduction to transpilation
- AI Transpiler service
- Transpiler plugins
- Qiskit ecosystem

Hackathon Prompt: Use Dynamic Circuits

Background and motivation:

Dynamic circuits are an exciting feature of IBM Quantum hardware that incorporates quantum circuits with real-time classical communication. Different from the static counterpart, dynamic circuits can not only implement a set of basic quantum operations like the Hadamard gate, CNOT gate, or qubit reset but also can implement measurement in the middle of a circuit, store the measurement results to classical bits, evaluate classical expressions on the fly, and determine what quantum operation to do next.
We encourage you to explore what you can do with dynamic circuits in this challenge! Some examples include but are not limited to:

- Find an application using dynamic circuits. *For example: prepare a large GHZ state, generate long-range entanglement, prepare a repetition code, do quantum phase estimation.*
- Demonstrate the improvement of implementation using the dynamic circuit feature. *For example: use dynamic circuit feature to shorten circuit depth or suppress error rate in a real circuit execution.*
- Characterize noise and errors in dynamic circuits. *For example, study how error propagates in dynamic circuits: use a benchmark method to study the fidelities of individual operations and composite operations in dynamic circuits.*
- With deeper understanding of how noise affects dynamic circuits, could you think of ways to suppress noise?

Deeper questions:

In the case of dynamic circuits, the types of error and how error propagates will be more complicated due to the hybrid quantum and classical nature of noise. We challenge you to characterize the noise in dynamic circuits. Tips: there is a suite of noise characterization and verification tools like tomography and randomized benchmarking, which are fairly well understood on static circuits, and extending them to dynamic circuits is an interesting research question to take on.

Suggested resources:

- What are dynamic circuits?
- Dynamic Circuits
- For testing and debugging the quantum circuits locally, qiskit_aer simulators can be useful
- Further reading about dynamic circuits:
  - Exploiting Dynamic Quantum Circuits in a Quantum Algorithm with Superconducting Qubits
  - Efficient Long-Range Entanglement using Dynamics Circuits
  - A randomized benchmarking suite for mid-circuit measurements
  - Randomized Benchmarking Protocol for Dynamic Circuits

Hackathon Prompt: Real-world Applications

Background and motivation:

We've entered a new era of quantum computing - The Quantum Utility era.

Quantum utility is what we get when a quantum computer can perform reliable computations at a scale beyond brute force classical computing methods that provide exact solutions to computational problems. Now, computational scientists and other researchers can tackle these large-scale problems using quantum computers with IBM's 100+ qubits backends accessible to everybody. Entering the era of quantum utility is, in other words, the quantum computers we have today are valuable tools researchers can use to explore meaningful scientific problems. Now it is your turn to tackle and explore issues with this leading-edge computational resource.

This challenge aims to design and build a cloud-based, quantum-powered application that addresses a real-world problem and is accessible to end users. This includes applications of quantum algorithms that can have practical usage and, in theory, be exposed to businesses or individual users on the internet for consumption. Teams should identify a problem that can be solved (though not necessarily more efficiently) with quantum computers.

Your prompt is to create a quantum-powered application that utilizes at least 80 qubits.

Deeper questions:

We encourage contestants to be creative and to leverage their existing quantum knowledge to develop new applications and experiments. To get you started, we offer some suggestions and prompts that may lead to interesting projects.

Some examples include but are not limited to:

- An application involving a Random Number Generator
- An application involving Optimization
- An application involving Chemical Simulation
- An application involving a Quantum Calculator
- An application involving Image Classification
- A game involving a quantum algorithm

Suggested resources:

- Variational algorithm design
- Solve utility-scale quantum optimization problems
- Quantum Computing in Practice playlist