1. Classification
1.1 MNIST Classification
Dataset: MNIST Dataset
Task: Build a classical ML model to predict the input images into final class labels. Calculate all the metrics such as accuracy, RoC, AUC, etc. Try different methods such as CNN, SVM, etc. to compare the results.

1.2 Use VQA algorithms to enhance the classical model. Build different hybrid models, combining both classical and quantum methods to test on the MNIST dataset. Calculate similar metrics for these quantum and hybrid models as well.

Bonus: Participants can try extending their work to more complex classification datasets such as the ChEMBL datasets for bioactivity classification or DDI (Drug-Drug Interation), PPI (Protein-Protein Interaction) tasks, etc.

2. Quantum RL
2.1 Classical RL
Train a classical RL agent to maximize reward for selected RL environment. eg. CartPole, FrozenLake, etc. Participants can refer to OpenAI Gym. Plot the training reward curves for the episodes. Try different algorithms such as Policy Gradient, PPO, DQN, etc. and compare the results.

2.2 Try implementing quantum and quantum-hybrid methods for action prediction, reward value functions, etc. Experiment with different designs and compare the results with the classical methods implemented. Explain why better final results or faster convergence are achieved. Keep in mind that Quantum models may *not* outperform classical models in terms of raw accuracy or training speed. Participants are encouraged to report negative results as well.

3. Generation Task
3.1 Build classical models for image generation. Try experimenting with different model designs such as VAE, VQ-VAE, GAN, etc. Experiment with the model designs and loss functions and show how these changes affects the different metrics.

3.1 Experiment with different quantum and quantum hybrid models to enhance the classical models. Try implementing models such as Quantum GANs and compare the results with the classical models. Plot all the results clearly and show how different variations of the model design and hyperparameters give different results. This could also include ablation studies.

4. Paper Implementation
Implement a Quantum Machine Learning paper of your choice. For paper implementation, the report should contain the reason for selecting the paper and to what extent the paper was implemented. The report should elaborate the structure proposed in the selected paper and how it was implemented. Results and plots should be clearly compared between the original paper and the implemented code. If a classical counterpart of the chosen model is already present, the

report should also contain comparisons between the classical models' parameters and results and those achieved from the implementations, this does not necessarily mean that the implementations outperforms the classical model design. If negative results are achieved, report the negative results as well with possible reasons.

Note: Quantum models may *not* outperform classical models in terms of raw accuracy or training speed. Participants are encouraged to report negative results as well — i.e., cases where quantum or hybrid approaches perform worse or offer no improvement. The primary goal is to analyze why this happens and to understand the current limitations of quantum machine learning (e.g., limited number of qubits, circuit depth, noise, feature encoding constraints, barren plateaus, etc.).

Deliverables

1. Report describing the different classical, quantum and quantum-hybrid models built and tested in detail. The report should contain the team name and names and discord usernames of team members.
2. Innovation used to build the models.
3. Training and Testing method description.
4. Comparison of all the metrics along with clear plots and diagrams.
5. Description of how the quantum model was run, with simulation details or description of hardware is circuit was run on real quantum hardware.
6. Submitted code should be reproducible
7. Report on limitations / negative results. Include discussion on whether the quantum/hybrid methods performed better or worse compared to classical models, and why. Clearly state observed limitations such as circuit complexity, encoding difficulty, optimization instability, or simulator noise effects. Negative results are equally valuable when well-explained.