Introduction to Quantum Computing with Azure Quantum and Q#

Course Description

This course introduces quantum computing with a focus on using Azure Quantum and Q# programming language. Participants will learn the fundamentals of quantum computing, including key concepts such as superposition, interference, and entanglement, and how to implement quantum algorithms using the Quantum Development Kit (QDK) and Azure Quantum services. The course will cover topics such as creating a quantum workspace in Azure Quantum, programming quantum algorithms with Q#, estimating physical resources for quantum algorithms using the Azure Quantum Resource Estimator, running quantum algorithms on quantum hardware, and solving optimization problems using quantum-inspired optimization. Through practical exercises and case studies, participants will gain hands-on experience in developing and running quantum programs with Azure Quantum and Q#..

Audience

This course is suitable for software developers, data scientists, and researchers who are interested in learning about quantum computing and want to get started with using Azure Quantum and Q# for quantum programming. It is also suitable for individuals with a background in computer science or a related field who want to explore the applications of quantum computing in various domains, such as optimization and algorithm design.

Pre-requisite Knowledge/Skills

Participants should have a basic understanding of programming concepts and some familiarity with a programming language, as well as basic knowledge of linear algebra and probability. Familiarity with quantum mechanics or quantum computing is not required, as the course will cover the necessary concepts.

Course Objectives

Upon completing this course, participants will be able to:

- Create a quantum workspace in Azure Quantum
- Write and run quantum programs using Q# and the Quantum Development Kit
- Understand key concepts of quantum computing, including superposition, interference, and entanglement
- Implement and run quantum algorithms, such as Grover's search, using Q# and Azure Quantum
- Estimate physical resources for quantum algorithms using the Azure Quantum Resource Estimator
- Utilize Q# libraries for quantum programming tasks
- Run quantum algorithms on quantum hardware using Azure Quantum services
- Solve optimization problems using quantum-inspired optimization techniques

Course Outline

The course comprises 24 hours of theory. It's divided into 9 different modules.

Module 1: Get started with Azure Quantum

- Introduction
- Azure Quantum structure overview
- Create your first Azure Quantum workspace
- Get started with a Jupyter notebook and Q# in Azure Quantum
- Case studies for quantum computing
- Case studies for optimization

Module 2: Create your first Q# program by using the Quantum Development Kit

- Introduction
- Exercise - Install the QDK for Visual Studio Code
- Exercise - Create a quantum random bit generator
- Exercise - Create a quantum random number generator
- How are Q# programs structured?

Module 3: Explore the key concepts of quantum computing by using Q#

- Introduction
- Superposition in quantum computing
- Exercise - Explore superposition by using Q# and Visual Studio Code
- Interference in quantum computing
- Exercise - Explore interference by using Q# and Visual Studio Code
- Entanglement in quantum computing
- Exercise - Explore entanglement with Q# and Azure Quantum
- Introduction to quantum algorithms

Module 4: Solve graph coloring problems by using Grover's search

- Introduction
- The search problem
- How to implement classical computation on a quantum computer
- Exercise - Implement a quantum oracle for graph coloring problem
- Grover's search algorithm
- Exercise - Implement Grover's algorithm to solve graph coloring problem
- Potential applications of Grover's algorithm in practice

Module 5: Estimate physical resources for quantum algorithms by using the Azure Quantum Resource Estimator

- Introduction
- Resource estimation in quantum computing
- The Azure Quantum Resource Estimator
- Exercise - Get started with the Azure Quantum Resource Estimator
- Exercise - Estimate resources for a real-world problem

Module 6: Use the Q# libraries

- Introduction
- Q# libraries
- Q# API documentation
- Exercise - Write an oracle to validate ISBNs
- Exercise - Run Grover's algorithm
- Exercise - Write your own documentation

Module 7: Run algorithms on quantum hardware by using Azure Quantum

- Introduction
- Quantum hardware overview
- Exercise – Submit a job to Azure Quantum
- Different targets in Azure Quantum
- Exercise – Solve a simple Grover's Search with Q# and an Azure Quantum notebook
- Continue experimenting with Azure Quantum

Module 8: Solve optimization problems by using quantum-inspired optimization

- Introduction
- Exercise - Implement a quantum-inspired optimization algorithm
- Exercise - Solve a combinatorial optimization problem
- Exercise - Solve a job shop scheduling optimization problem

Module 9: Advanced Topics

- Intermediate and Advanced concepts in quantum computing and optimization