AMBIENT-IT

+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 27/05/2024

Sign up

Q# training with Azure Quantum

3 days (21 hours)

Presentation

Q# (or Q Sharp) is the reference quantum language, enabling you to solve ultra-complex problems thanks to new scientific advances. Quantum programming could revolutionize many sectors, from health to security.

Launched by Microsoft on September 11, 2017, this language used to simulate quantum algorithms, is inspired by F# and C#. The technology can be used to manage qubits (the quantum computing unit, the equivalent of the bit in binary language), develop on first-class objects, derive operations and develop quantum circuits.

Q# is launched thanks to Azure Quantum, a Cloud service that lets you run programs using quantum technology. Azure Quantum is an open ecosystem, giving you access to various quantum solutions from Microsoft and its partners.

During our Q# with Azure Quantum training course, you'll learn how to develop in Q# on the Azure Quantum environment, create quantum programming algorithms and optimize your Azure Quantum infrastructure.

Our Q# training course will introduce you to the latest version of the Quantum Development Kit, QDK 0.18.2109.162713.

Objectives

- Learn how to configure and optimize Azure Quantum
- Creating algorithms in Q#
- Developing non-local games
- Moving quantum data with teleportation and entanglement

Target audience

- Developers
- Engineers

Prerequisites

- Development experience, ideally with Microsoft programs
- Our quantum programming training is recommended
- An Azure account with Quantum functionality

Our Q# with Azure Quantum training program

Introduction

- A reminder of how quantum computing works
- The benefits of quantum programming in the 21st century
- Q# application cases
- QDK installation and configuration

AZURE QUANTUM: QUANTUM CLOUD

- Configuring Azure Quantum
- Running quantum computing programs
- Discover the QIR (Quantum Intermediate Representation) intermediate language
- Learn how to build and test quantum applications in the Azure cloud
- Q# User Guide
- Update QDK

Coding in Q#

- Types
- Declarations
- Expressions
- Operators
- IQ controls
- The functions
- Developing a game in Q#

NON-LOCAL GAMES: WORKING WITH MULTIPLE QUBITS

- What is a non-local game?
- Working with multiple qubit states
- Classic strategy
- Difficulty simulating quantum computers

- The registers
- Tensor products for qubit operations on registers
- Troubleshooting

NON-LOCAL GAMES: IMPLEMENTATION OF A MULTI-QUBIT SIMULATOR

- Quantum objects in QuTiP
- Simulator update
- How do you measure multiple qubits?
- CHSH: Quantum strategy
- Debugging

TELEPORTATION AND ENTANGLEMENT: MOVING QUANTUM DATA

- Quantum teleportation
- Moving quantum data
- Change simulator
- What other two-qubit gates are there?
- All qubit rotations
- Relating rotations to coordinates : Pauli operations

Optimizing Azure Quantum

- Quantum-inspired optimization (QIO)
- Solving optimization problems in the cloud
- Publish a QIO job as an Azure function
- Binary optimization (QUBO and PUBO)
- Cost function and Ising model
- Azure Quantum with Azure CLI
- Troubleshooting

Companies concerned

This training course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or internal security difficulties within the company (intra-company or virtual classroom) that might be encountered.

problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical course: 60% Practical, 40% Theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Sanction

A certificate will be issued to each trainee who completes the course.