Setting Up a Programming Environment

Understanding the theory behind quantum computing is critical to leaving your mark on the field, however, it isn't possible without having some implementation experience under your belt.

You might say, "This is all well and good, but I don't have a quantum computer!". You needn't worry, IBM has got your back. IBM has created a software library called 'Qiskit' which allows people studying and researching quantum computation to not only simulate quantum computers on their local machines to test quantum circuits, but also run their code on actual quantum computers!

Let's get started with Qiskit!

1 Prerequisites

Before we can get started with using Qiskit, we need to install all the other software it depends on. Setting a programming environment can vary greatly between operating system – we shall try as operating system agnostic as possible, but will go into detail should a particular step be different for an operting system.

1.1 Python

Qiskit is written in the Python programming language, so it goes without saying that we'll need to have Python installed on our machine. If you already have python installed, you can skip to the next step.

We recommend that you install the Anaconda distribution of Python from this link.

1.2 Dependencies

Now that we have Python installed we can start installing the dependencies.

If you have installed an anaconda distribution of Python, then you need only open the anaconda prompt and type the following command:

```
$ conda env create -f /path/to/requirements.yml
```

You should now have a virtual environment called `quantum` with Qiskit along with all its dependencies installed.

If you already had Python installed, then you must create a new virtual environment with `python v3.7`, and install `pip`, `matplotlib`, `notebook`, `ipywidgets`, `rise`, and finally, `qiskit`.

2 Qiskit

Now that we have Qiskit installed, we must set up our machine to be able to run our circuits on IBM's quantum computers using the IBMQ provider.

To access IBMQ devices, you need an API token, generated here, after you have made an account.

Fire up a terminal, and run the following lines of code in an interactive Python shell:

```
>>> from qiskit import IMBQ
>>> IBMQ.save_account("YOUR_TOKEN")
```

You are now ready to go!