Hi!

Once again, thank you for your interest in the QC Mentorship program!

We decided to select participants based on how they will manage to do some “test tasks”.

These tasks have been designed to:
- find out if you have the skills necessary to succeed in our program.
- be doable with basic QC knowledge - nothing should be too hard for you to quickly learn.
- allow you to learn some interesting concepts of QC.
- give you some choice depending on your interests.

What we understand by skills is not knowledge and expertise in QC. It’s the ability to code, to learn new concepts and to meet deadlines.

Be creative. While solving simpler versions of these is not a good idea, after solving a basic version you can get creative.

Choose task based on your interests, don’t try to pick the easiest one.

You need to do only 1 task. Feel free to do all of them, it might be a good learning opportunity, but it won’t affect admissions to the program :)

So here are the tasks:

Task 1
Solve MaxCut problem on a weighted graph using grove (https://github.com/rigetti/grove/tree/master/grove/pyqaoa).

Then run it on a qvm with and without noise and describe your observations.

Since what’s implemented in grove does not support weighted graphs, finishing this will require introducing changes to the source code of grove.

Task 2
Implement a circuit which returns |00> and |11> with equal probability.
Requirements :
- Circuit should consist only of CNOTs, RXs and RYs.
- Start from all parameters in parametric gates being equal to 0 or randomly chosen.
- You should find the right set of parameters using gradient descent (you might use more advanced optimization methods if you like).

- Simulations must be done with sampling - i.e. limited number of measurements per iteration and noise.

Compare the results for different numbers of measurements: 1, 10, 100, 1000.

Bonus question:
How to make sure you produce state |00>+|11> and not |00>-|11> ?

Task 3
Write script which takes as inputs:
- quantum program (e.g. in QASM or QUIL)
- errors/noise parameters (coherence time, 1&2 qubit errors, measurement error)
- or no errors

and calculates probability of getting a correct result.

Some requirements:
- Use realistic values for the parameters.
- You don't have to support all gates - just enough to run QFT.

Test it on QFT (Quantum Programming Studio is a good tool for generating input circuits).

Task 4

10 Feb update: I changed the matrix slightly - added the minus sign compared to the original one. This should make the solution less degenerate.
10 Feb 2nd update: Ok, I've double checked it and this is indeed a reasonable problem with reasonable tips :D Sorry for confusion!

Find the lowest eigenvalue of the following matrix:

[0 0 0 0;
0 -1 1 0;
0 1 -1 0;
0 0 0 0]

using VQE-like circuits, handwritten by yourself.

In general this exercise might be pretty difficult, so below there are a couple of tips. I've written them in pale font, so that those of you who embrace the challenge don't look at them accidentally ;)

1. It requires decomposing the matrix to sum of Pauli terms.

2. Decomposition involves only terms consisting of the same matrices, i.e.: II, XX, YY, ZZ and the coefficients are from the set [-2, -1, 0, 1, 2].
3. The ansatz you can use is: (IX) CX (RZ I) (HI) |00>, where angle in RZ is your variational parameter.
4. You can find an explanation of VQE in this blogpost. You can also find links to further resources there.
5. You can just search through all angles for RZ, you don't need to use any optimizers like gradient descent.

Deadline
2 weeks.

Once you have finished, please put your code on GitHub and send us a link to the repository at michal@qosf.org – other forms of submission won't be accepted!

If you have any questions - please add comment to this document, we will be updating this document with more details and/FAQ to avoid confusion, so make sure to check it before asking :)

Have a nice day!
Michał Stęchły

FAQ

Q: Can we use any quantum libraries or are we restricted to a particular set of tools?

Feel free to use whatever you like, just make sure that the tool doesn't solve the whole problem for you.
Not open source packages like Mathematica or Matlab or ok-ish, but if you can do that in Python it is definitely preferred. You can do it first in the language of your preference and then translate it to Python - I think that would be optimal.

Q: I found some issues with Task 4 and ....
I think it should be good now. But if you still have doubts and something doesn't add up, please let me know :)

Q: How should I submit the solution?
All the materials for the submission should be inside the GitHub repository. Please do not send us any loose files as attachments or in any other format. Link to the GitHub repo in the e-mail to michal@qosf.org is all we want.

Q: What is ETC timezone

In the e-mail I've sent you information about the deadline with the ETC timezone, which does not exist.

I meant EST, sorry for confusion!