Shor's Algorithm

Elliott Pryor, Benjamin Bushnell, Shengnan Zhou

25 November, 2019

Our Algorithm of study is Shor's algorithm, and our +1 is to walk through an example run of a classical version of Shor's algorithm.

1 What we've done

We figured out pretty quickly how to run through an example of Shor's algorithm using classical computation. As shown in section 5 of our submission of P-4, we have already demonstrated that we can run through a simple example of Shor's. For convenience, here it is again:

To better understand how Shor's algorithm works, let's walk through an example.

Let's say $N = 21$, and we want to find a, b such that $N = a \cdot b$ and a, b are prime numbers. First, we take a random guess $g = 11$. We know that the GCD of 11 and 21 is 1, so we proceed to turn g into a better guess $g^{p/2} \pm 1$. Now we want to find p such that $g^p = m \cdot N + 1$ for some m.

We can find p by calculating the period at which a^{x+yp} (mod N) repeats. On a quantum computer this is very fast; however, we can easily do this by hand as 21 is a very small number (and our power is also small). Below, is 11 raised to the powers 0-19 (mod 21).

$$1, 11, 16, 8, 4, 2, 1, 11, 16, 8, 4, 2, 1, 11, 16, 8, 4, 2, 1, 11$$

We can see that the cycle repeats every 6 iterations. Therefore, we know that $p = 6$.

We find $11^6 = 84360 \cdot 21 + 1$. Now we can have the better guess $g^{p/2} + 1 = a \cdot s$ for some factor s, and $g^{p/2} - 1 = b \cdot t$ for some factor t. To find a, b, we need to use Euclid's algorithm to find the common factors. $g^{p/2} + 1 = 1332$, $gcd(1332, 21) = 3$. And $g^{p/2} - 1 = 1330$, $gcd(1330, 21) = 7$.

Now we found the two prime factors of $N = 21$ are $a = 3$ and $b = 7$.

In P-4, we outlined exactly what needed to be coded for our +1 to be implemented:

1. Euclid's Algorithm for GCD/finding the guess
2. Period finding

We have addressed both of these items in a classical adaptation of Shor's written in Python. To make sure it runs as expected, we have tested it with a few numbers. Most of our tests pass, but the program has some bugs that need ironing out, specifically when we test it on larger numbers.

2 What we have left to do

First and foremost, we need to debug the python program, and choose a specific example to walkthrough. Once we have those things, we then need to determine the format and outline of our video. This is important, because if we don't do a good job with the outline, it will be difficult for the viewer to understand our project. We still have quite a bit of work left to do regarding this, but we have some ideas for a few elements of the video. Likely, we will film one of our members drawing an example walkthrough of classical Shor's with narration over the top. Furthermore, we may also use screen capturing to show how our python implementation works. We will film an introduction which will provide context for the viewer to understand our +1, and may discuss how Shor's affects modern encryption. Once we've decided how all of these things fit into the video outline, it will just be a matter of filming and editing the final video.