Section I

Part one

1) The no-cloning theorem of quantum computing states that it's impossible to create an identical, independent copy of an arbitrary unknown quantum state. We can interpret this, by considering an arbitrary function U_x acting on one input $|\psi\rangle$. The no-cloning theorem states that we cannot have the function output two $|\psi\rangle$. This happens, because converting a basis qubit to an arbitrary qubit is not a linear transformation, while the function U_x is linear, and so can only perform a linear transformation.

   Another way to think of this is, consider a machine which randomly shoots out a colored ball from 20 set colors. If we put this machine into a closed box, and don't look inside it, we cannot perfectly recreate the box, especially one which will give the exact same randomization. The only way to replicate it would be to look into the box and see what colour and what randomization the machine uses, but then the randomness of the first block will be destroyed to create a new randomized box, so it wouldn't be cloned, but rather copied.

2) For an Operation to be a valid Quantum gate, it must be Unitary. If, for a function U, we can say that the statement $U^\dagger U = I$, where $U^\dagger$ is the complex conjugate transpose of the matrix U, and I is the identity matrix, then we can say that the matrix U is unitary.

   By being unitary, we can say the function U is also reversible and linear, since by fulfilling the criteria to be unitary, U also naturally has an inverse, and it is also linear, meaning it can act on linear combinations of quantum states.

3) A projective measurement is the measurement of the quantum state in such a way that the quantum particle's value which we are measuring collapses onto a single value, and any following measurements will always give the exact same value, as by measurement, we will have made the probability of the measured value appearing as 100%. On the other hand, a quantum operation affects the value and state of the qubit, without actually measuring it, therefore making sure that the qubit does not collapse onto any one value.

   An example of a quantum operation is the Pauli-X gate or the X gate. This gate acts as a quantum NOT gate, effectively changing the phase of the qubit by 180°, or rotating about the X-axis by 180° on the Bloch sphere. On the other hand a projective measurement is a process in which we read the value of the qubit through its projections upon the basis vectors, thus collapsing the qubit onto one state.

4) Quantum Entanglement is the phenomenon by which two quantum particles are associated directly, in such a way that even over large distances we cannot define the state of one particle without the state of the other particle.

   A physical example of two systems whose states are correlated such that measuring one instantly determines the state of the other, regardless of distance, is if we randomly choose one of two boxes, and put a black cube within it, and put a white cube in the other. Now, if we give them to two people, who know that the boxes either contain a white cube or a black cube, and have them travel very far away from each other, then we can have one of the people open their box, and depending on the colour of their cube, they will instantly know the color of the other person's cube, regardless of how far apart they are.

Quantum entanglement differs from perfect correlation in the following way. In Quantum entanglement, if we take two entangled particles, and split them far apart, then we can not only know the value of the other particle by measuring the particle in our possession, but we can also force the value of the other particle into a certain state, by manipulating the associated value of the particle in our possession prior to the observation, while in perfect correlation, this latter part cannot be performed i.e. we cannot force the value of the other particle to change by affecting the particle in our possession before observation.

Part two

1) Given: $U = X \otimes H$, where X is the Pauli-X gate and H is the Hadamard gate.

We know, Matrix representation of X and H

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \text{ and } H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$\begin{aligned} U &= X \otimes H \\ &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ &= \begin{bmatrix} 0 \times \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & 0 \times \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \end{bmatrix} \\ &= \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \\ 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix} \end{aligned}$$

Therefore, the explicit matrix representation of U is $U = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \\ 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix}$

2) Given V = (H ⊗ I) · CNOT · (X ⊗ Y), where H is the Hadamard, X and Y are Pauli gates, I is the identity, and a CNOT b is the NOT (Pauli-X) gate controlled on qubit a and applied to qubit b, if the initial state is |01⟩, what is the final state after applying V?

We know, Matrix representation of X,Y,I,CNOT and H

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

And Lastly, $\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

First, we calculate $H \otimes I$

$$H \otimes I = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{bmatrix} 1\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & 1\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 1\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & -1\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad \text{-------(1)}$$

Next, we calculate $X \otimes Y$

$$X \otimes Y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 1\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ 1\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 0\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & 1 & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix} \quad \text{-------(2)}$$

Initial State → $|01\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ Let this be called $|\psi\rangle$

The operation to be performed upon $|\psi\rangle$ can be written as: $(H \otimes I \cdot CNOT \cdot X \otimes Y)\,|\psi\rangle$

Performing in parts:
Part 1: $(X \otimes Y)\,|\psi\rangle$

$$(X \otimes Y)\,|\psi\rangle = \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & 1 & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0 \\ 0 \\ -i \\ 0 \end{bmatrix} = -i \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = -i|10\rangle$$

Part 2: $(CNOT) \cdot -i|10\rangle$
$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot -i \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$
$$= -i \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$
$$= -i|11\rangle$$

Part 2: $(H \otimes I) \cdot -i|11\rangle$
$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \cdot -i \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$
$$= \frac{-i}{\sqrt{2}} \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \end{bmatrix}$$
$$= \frac{-i}{\sqrt{2}} \left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right)$$

$$= \frac{i}{\sqrt{2}}(|01\rangle - |11\rangle)$$

Here, the i demonstrates the phase, so we can just forget that it ever existed, so our answer can look pretty :D

So, The final state after applying V on initial state $|01\rangle$ can be denoted by

Final state $|\psi_1\rangle = \frac{i}{\sqrt{2}}(|01\rangle - |11\rangle)$

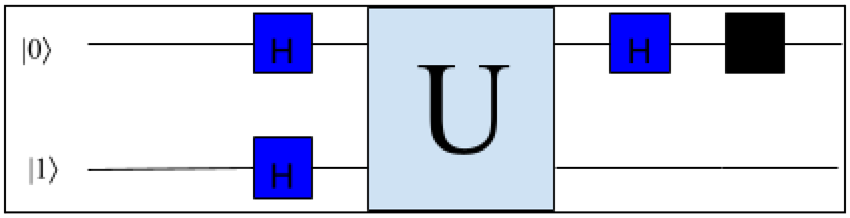
Section II

PS2

Part A: Theoretical

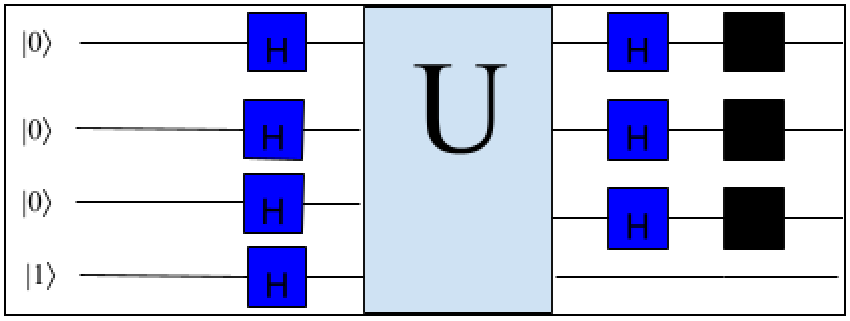
The first thought was to go through exactly half of the inputs, so 4, and see if they gave all the same answers. If they did, it would be a constant function, if they didn't, the moment we would see a different output we could say that it was a balanced function. We could say this because we only have two possible cases, either completely constant, or exactly balanced. This was slightly wrong as we actually needed to do this one more than exactly half, so 5 times, since the first 4 could be 0 and the last four could be 1, and we wouldn't catch them.

Then I realised that I had an IQ in the single digits, and this was a classical code, with nothing in quantum. Then I watched the FreeCodeCamp video and saw the deutsch algorithm. I then had the following ideas for extending it to multiple qubits since I wanted to try it for myself.



Model for Deutsch's Algorithm for single qubit system

The first (And completely arbitrarily thought of and done) idea was to directly add two more lines for the zero qubit, while leaving the bottom one qubit untouched.



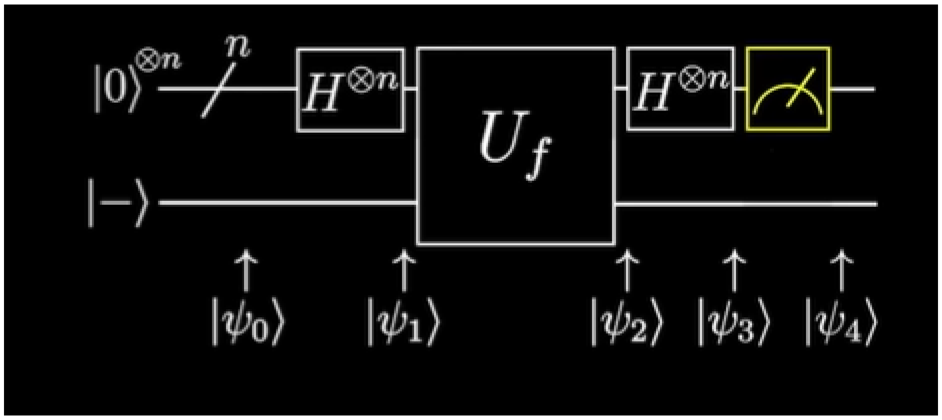
First Idea (IDK how but it worked :D)

Then, similar to Deutsch algorithm, (or I guess identically since nothing was added), I solved for the state $|\psi\rangle$ at 1,2,3 and output. Image of calculations is attached (I got too tired at this point to typeset anything more).

Here, I got confused on how to solve further for the Hadamard Gate and get the answer. Then I found a way to think of x as $x_1x_2x_3$, and then apply the Hadamard Gate separately.

So what was found, that in fact this model worked. If the function f was balanced, then the measured result would be anything other than the $|000\rangle$ state, But if the function was constant, then the measured state would definitely be the $|000\rangle$ state.

Then I actually looked forward in the video, which I had used for help in some of the Math, and found that this was the exact result used, and this algorithm was called the Deutch-Jozsa Algorithm.



Deutsch-Jozsa Algorithm for n-qubit system

While the algorithm is solved for the general case of n-qubits, for our case, we would use the n=3 case, giving the exact same result.

Now that this was done, ~~I went and played Minecraft for 3 hours straight.~~ I decided to write a Code using Quiskit to represent this.

Part B: Coding

"We are in a hell of our own making" - Kim Dokja

I thought it would be simple. I was stupid. I started by defining a function that had two ways, one in which you could either choose to input the function yourself or have it automatically made. The automatically made had an equal probability of being either perfectly balanced, or fully constant, with both 0 filled or 1 filled. Then when I tried to integrate it into the rest of my circuit, I could not understand how to have every index of the function act on the states one by one. So I opened the quiskit implementation page on IBM, and realised **I did everything wrong and that I needed to delete everything and start again.** ~~**I Then Screamed Into A Pillow**~~ I then consulted the IBM Quiskit implementation pages and created a new oracle, Then defined a CircuitMaker, created the Hadamard,Then Oracle, Then Hadamard Circuit. Then I defined the Result Giver Function which would compute the resulting string and output if the function was balanced or constant.

All this was written in Notepad since I did not have enough space in my laptop for an IDE, so there might be many errors. Please don't bully me because of the hundreds of errors that would be present, and look at the logic instead. Thank You.