Name- Jitendra kumar

Roll- CrS1914

~~S~~Sub: Quantum Cryptology and security

——— × ——— × ——— × ——— × ——— × ———

SOLUTION:-

① a) Since the tensor product of n 2-dimensional unit vector space gives us 2^n-dimensional unit vector space.

So, In this way, We can construct a n-qubit state.

For example,

Let
$$|v_1\rangle = \alpha_1|0\rangle + \beta_1|1\rangle$$
$$|v_2\rangle = \alpha_2|0\rangle + \beta_2|1\rangle$$
$$|v_3\rangle = \alpha_3|0\rangle + \beta_3|1\rangle$$
$$- \quad - \quad \cdot$$
$$|v_n| = \alpha_n|0\rangle + \beta_n|1\rangle$$

where ~~⊗~~ $|\alpha_i|^2 + |\beta_i|^2 = 1$

Now, for constructing n-qubit state, We take the tensor product as.

$$|v\rangle = |v_1\rangle \otimes |v_2\rangle \otimes \cdots \otimes |v_n\rangle$$

$$= \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \\ \delta_n \end{bmatrix} = \delta_1 \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \delta_2 \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \cdots + \delta_n \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \\ 1 \end{bmatrix}$$

where $|\delta_1|^2 + |\delta_2|^2 + \cdots + |\delta_n|^2 = 1$.

and δ_i are complex numbers.

Two Examples are as:

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}_{2^n \times 1} \quad \text{and} \quad \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ \\ 0 \end{bmatrix}_{2^n \times 1}$$

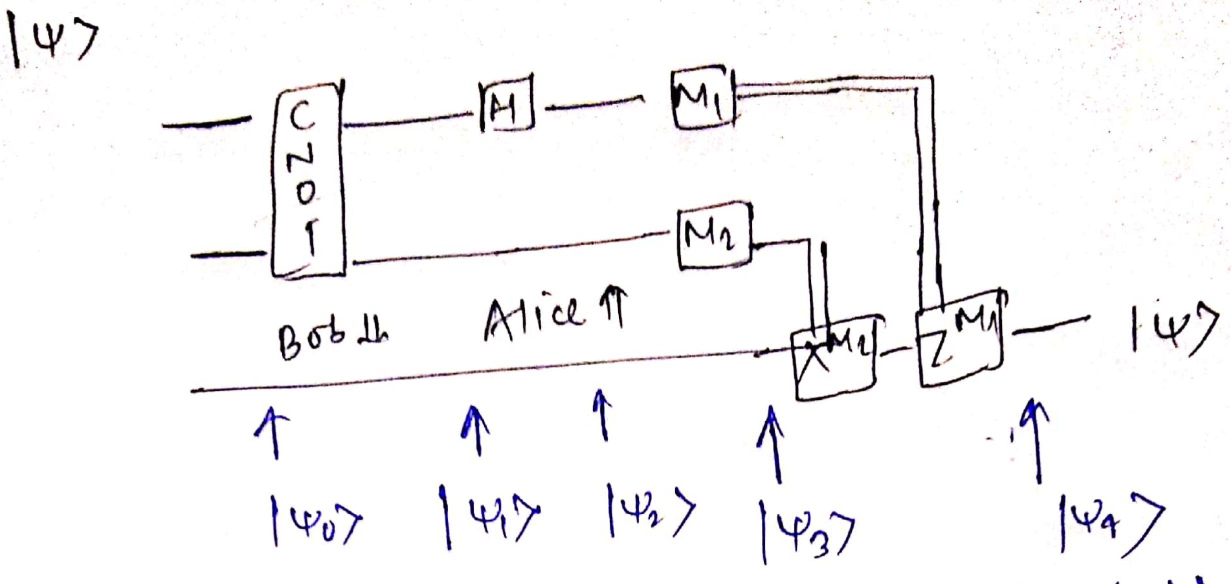
① ⓑ An n-qubit input, n-qubit output quantum gate is precisely $2^n \times 2^n$ unitary matrix. ie $\boxed{U\bar{U}^T = I}$

Example.

(i) $U = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & & 1 & 0 \\ & - & - & \cdot & \\ 0 & 1 & - & 0 & 0 \\ 1 & 0 & - - & 0 & 0 \end{bmatrix}_{2^n \times 2^n}$

(ii) $U = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & - - & 0 \\ - & & & — & \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}_{2^n \times 2^n}$.

②



$$|\psi_0\rangle = |\psi\rangle|\beta_{00}\rangle = (\alpha|0\rangle + \beta|1\rangle)\frac{(|00\rangle + |11\rangle)}{\sqrt{2}}$$

$$|\psi_1\rangle = \alpha\,|0\rangle\frac{(|00\rangle + |11\rangle)}{\sqrt{2}} + \beta\,|1\rangle\frac{(|10\rangle + |01\rangle)}{\sqrt{2}}$$

$$|\psi_2\rangle = \alpha\left(\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)\left(\frac{(|00\rangle + |11\rangle)}{\sqrt{2}}\right)\right.$$
$$+\beta\left(\frac{|0\rangle \cancel{+} - |1\rangle}{\sqrt{2}}\right).$$
$$\left.\left(\frac{(|10\rangle + |01\rangle)}{\sqrt{2}}\right)\right)$$

$$= \frac{1}{2}|00\rangle(\alpha|0\rangle + \beta|1\rangle)$$
(Nothing to do)

$$+ \tfrac{1}{2}|01\rangle(\beta|0\rangle + \alpha|1\rangle)$$ (apply X-gate)

$$+ \tfrac{1}{2}|10\rangle(\alpha|0\rangle \cancel{+} \beta|1\rangle)$$ (Apply Z-gate)

$$- \frac{1}{2}|11\rangle(\beta|0\rangle - \alpha|1\rangle)$$ (Apply both X & Z gates)

Solution ④

GROVER'S ALGORITHM!

a)

Grover's Algo. is used as searching entry from an unstructured database of size N with $O(\sqrt{N})$ queries.

Problem statement! -

Given a function

$$f: \{0,1\}^n \rightarrow \{0,1\}$$

The goal is to find $x \in \{0,1\}^n$, such that $f(x)=1$ or to conclude that no such x exists. ie $f=0$, a constant function.

Let's assume

$$A = \{ x \in \{0,1\}^n : f(x)=1 \}$$

$$B = \{ x \in \{0,1\}^n : f(x)=0 \}$$

Also, assume $|A|=a$, $|B|=b$

for $N=2^n$, $a+b=N$

Initialization

→ Begin with a state $|\psi_0| = |0\rangle^{\otimes n}$

matrix Representation:

$$|\psi_0| = \begin{pmatrix} 1 \\ 0 \end{pmatrix}^{\otimes n}$$

→ Apply the Hadamard gate to each of these qubits.

$$|\psi_1\rangle = (H|0\rangle)^{\otimes n}$$

$$|\psi_1\rangle = \left(\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right)^{\otimes n}$$

$$= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$

$$= \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} |x\rangle$$

Consider the states!

$$|A\rangle = \frac{1}{\sqrt{a}} \sum_{x\in A} |x\rangle \quad \text{and} \quad |B\rangle = \frac{1}{\sqrt{b}} \sum_{x\in B} |x\rangle$$

~~Since~~ Note that $|A\rangle$ and $|B\rangle$ are orthogonal.
Consider the space spanned by $|A\rangle$ and $|B\rangle$.

Thus, $$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x\in\{0,1\}^n} |x\rangle = \frac{1}{\sqrt{N}} \left(\sum_{x\in A} |x\rangle + \sum_{x\in B} |x\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \left(\sqrt{a} \times \frac{1}{\sqrt{a}} \sum_{x\in A} |x\rangle + \sqrt{b} \times \frac{1}{\sqrt{b}} \sum_{x\in B} |x\rangle \right)$$

$$\Rightarrow |\psi_1\rangle = \sqrt{\frac{a}{N}} |A\rangle + \sqrt{\frac{b}{N}} |B\rangle$$

Assuming $\sqrt{\frac{a}{N}} = \sin\alpha$ & $\sqrt{\frac{b}{N}} = \cos\alpha$.

$$\Rightarrow \alpha = \sin^{-1}\left(\sqrt{\frac{a}{N}}\right)$$

$$Z_f|x\rangle = (-1)^{f(x)} |x\rangle$$

$$Z_0|x\rangle = \begin{cases} -|x\rangle & \text{if } x = 0^n \\ |x\rangle & \text{if } x \neq 0^n \end{cases}$$

Matrix Representation:

$$Z_0 = 1 - 2|0^n\rangle\langle 0^n|$$

Now, for $$G|A\rangle = [1 - 2|\psi_1\rangle\langle\psi_1|](-Z_f)|A\rangle$$

$$= [1 - 2|\psi_1\rangle\langle\psi_1|]|A\rangle$$

$$= |A\rangle - 2\sqrt{\frac{a}{N}}\left(\sqrt{\frac{a}{N}}|A\rangle + \sqrt{\frac{b}{N}}|B\rangle\right)$$

$$= \left(1 - \frac{2a}{N}\right)|A\rangle - \frac{2\sqrt{ab}}{N}|B\rangle$$

Similarly, $$G|B\rangle = \frac{2\sqrt{ab}}{N}|A\rangle - \left(1 - \frac{2b}{N}\right)|B\rangle$$

Thus, G can be considered as a matrix:

$$\begin{pmatrix} -\left(1 - \frac{2b}{N}\right) & -\frac{2\sqrt{ab}}{N} \\ \frac{2\sqrt{ab}}{N} & \left(1 - \frac{2b}{N}\right) \end{pmatrix}$$

Now. using $N = a+b$, we get

$$G = \begin{pmatrix} \frac{b-a}{N} , & -\frac{2\sqrt{ab}}{N} \\ \frac{2\sqrt{ab}}{N} & \frac{b-a}{N} \end{pmatrix}$$

$$= \begin{pmatrix} \sqrt{\frac{b}{N}} & -\sqrt{\frac{a}{N}} \\ \sqrt{\frac{a}{N}} & \sqrt{\frac{b}{N}} \end{pmatrix}^2$$

$$= \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}^2$$

Thus, G can be considered as Rotation matrix. which on application to a state, increase it angle by 2θ.

And our goal is to increase the probability of getting $|A\rangle$ ie $\sin\theta \approx 1$

Solⁿ:

(4) (b) Comparison with classical Algorithm

→ In classical Algo. We make 2^n many queries to the black box (in the worst case) to check if f is identically 0 or any of the cases exist where f outputs 1. Probabilistically though we can choose k many distinct values of x and make queries for those k values. This Algorithm succeeds with probability $1-\varepsilon$ where $1-\frac{k}{2^n} \leq \varepsilon$ ie the complexity is $\Omega(2^n)$ anyway.

On the other hand, the Grover's Algo. We can solve the same problem with $O(2^{n/2})$ complexity in quantum scenario.

⑤ a) <u>BB84 Quantum Key Distribution Protocol</u>

<u>First stage of Protocol:</u>

Alice randomly generates two strings of bits $x, y \in \{0,1\}^n$

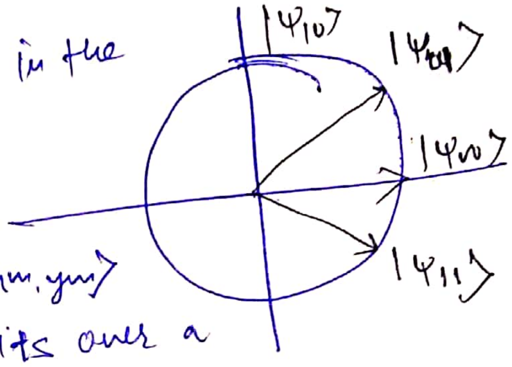
Define $|\psi_{00}\rangle = |0\rangle$

$|\psi_{10}\rangle = |1\rangle$

$|\psi_{01}\rangle = |+\rangle$

& $|\psi_{11}\rangle = |-\rangle$

We have these 4 states as:



Alice prepares m q-bits in the state

$|\psi_{x,y}\rangle$

$= |\psi_{x_1 y_1}\rangle \ldots |\psi_{x_m, y_m}\rangle$

and sends these m q-bits over a quantum channel to Bob.

Bob receives m-bits, ~~over a quantum channel~~ Although they may not longer be in a state $|\psi_{x,y}\rangle$ because Eve may have tampered with them, or possibly the channel is noisy.

Bob randomly chooses $y' \in \{0,1\}^m$ and ~~means~~ measures each q-bit recieved from Alice as follows!

- If $y^i = 0$, Bob measures qubit.
- If y_i; Bob performs a Hadamard transform to q-bit i and then measures it with respect to the standard basis.

Let $x' \in \{0,1\}^m$ be the string corresponding to the results of Bobs, measurements. The important thing to note at this point is that if $y_i = y_i'$ for some i and there was no noise or eavesdropping, then It is certain that $x_i = x'$

Finally, Alice and Bob Publicly Compare y and y'. They discard all bits x_i and x'_i for which $y_i \neq y'_i$. The Remaining bits of x and x' represent a 'Semi-private' key that will be go into the next stage of the protocol.

2nd stage of protocol:

Alice and Bob now need to estimate how much Eve might know about x and x'. They do this by Scarificing Some of. They do this by Scarificing Some of bits of x and x'. By comparing these bits publicly, they can estimate the error rate with high accuracy, and if it is too large they abort. This maximum error rate that can be tolerated is about 11%. If they have acceptable Rate error Alice and Bob will have ~~too~~ two strings x and x' that agree in high Percentage of positions with high Prob. They have some bound on the amount of info. Eve posses about these Strings.

Solution:

(5) (b) We again consider the scenario where the communicated qubits are intercepted by an eaves dropper and ~~[illegible]~~ measured in some orthogonal basis. Note that, the security of the protocol is based on the fact that if one wants to distinguish two non-orthogonal quantum states, then obtaining any information is only possible at the expense of introducing disturbance in the state.

Suppose, Alice communicated in Standard basis. Now, Eve can copy that perfectly without creating any disturbance to $|\psi\rangle$. ie passed between Alice and Bob. We know cloning is possible for orthogonal vectors. Hence the bit error rate in this case will be 0 between Alice & Bob and Eve's success probability will be 1 using the CNOT Attack.

If Alice communicates in Hadmard Basis, then output of CNOT gate would be entangngled states. Hence Bob, if measures in Hadmard basis will observe the $|+\rangle$ and $|-\rangle$ values with probability $\frac{1}{2}$. ie bit error rate in this case is $\frac{1}{2}$ between Alice and Bob and success probability of Eve would be $\frac{1}{2}$ also. This will lead to any metric ~~Eave~~ Evesdropping

Solution:

③ (a) Deutsch _Jozsa Algorithm

Problem statement: Suppose f be a boolean function $f: \{0,1\}^n \to \{0,1\}$. The property of the given f is that It is guaranted to either be balanced or Constant. (A Constant function returns all 0's or all 1's while Balanced function function return 0's for exactly half inputs and 1's for other half inputs).

Deutsch- Jozsa Algo. determines If the given function is balanced or Constant

③ (c) To solve the stated problem in question 3(a). The classical way to approach the solution is following. We choose the ~~outputs~~ inputs $|q_0 q_1 .. q_{n-1}\rangle$ with $q_i \in \{0,1\}$. And for every Such input of first registers, we check what f outputs. If f outputs 1 everywhere or 0 everywhere We claim that this is constant. Since f is either constant or Balanced, If f outputs 1 for some input and 0 for some other, f is necessarily balanced.

Hence the ~~Classical~~ Classical Algo. takes all 2^n inputs. In deterministic classical Algo., in the worst case We might have to check more than half of the values. ie $2^{n-1}+1$ queries might be needed.

With our Deutsch_Jozsa Algo. We can conclude the result deterministically with Just a single query to the oracle. Hence In contrast to Classical paradigm, Deutsch-Jozsa determines the balanced/constant property of f much quicker.

③ ⑥ Deutsch - Jozsa Algorithm

→ Prepare two quantum registers, the first one is an n-qubit quantum register with all the qubits are initialized to $|0\rangle$. And the second one is 1-qubit register initialized to $|1\rangle$.

$$|\psi_0\rangle = |0\rangle^{\otimes n} |1\rangle$$

→ Then Apply Hadamard Gate to each qubit

$$|\psi_1\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x \in \{0,1\}^n} |x\rangle(|0\rangle - |1\rangle)$$

→ Apply the quantum oracle U_f:

$$|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle \text{ on } \psi_1\rangle \,!$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x \in \{0,1\}^n} |x\rangle(|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle)$$

$$\Rightarrow |\psi_2\rangle = \frac{1}{\sqrt{2^{n+1}}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle(|0\rangle - |1\rangle)$$

→ Ignore the last qubit from the second Register, and apply Hadamard Gate to all the n qubits from the first Register.

$$H^{\otimes n}\left(\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}|x\rangle\right)$$

$$=\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\left[\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle\right]$$

$$=\frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}\left[\sum_{x\in\{0,1\}^n}(-1)^{f(x)}(-1)^{x\cdot y}\right]|y\rangle$$

→ Measure all the n-qubits from the first Register:

→ Probability of getting the state $|0\rangle^{\otimes n}$ is equal to $\frac{1}{2^{2n}}\left[\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\right]^2$

→ (i) If f(x) is a constant, then check that the above probability is 1.

(ii) If f(x) is balanced, the probability is 0.

③ (d) Given $f : \{0,1\}^3 \to \{0,1\}$

$$(x_1, x_2, x_3) \mapsto x_1 x_2 x_3$$

As described in the Algo.

$$|\psi_0\rangle = |000\rangle \otimes |1\rangle$$

Then $|\psi_1\rangle = \frac{1}{2\sqrt{2}} \{ |000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle \} \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

$$\Rightarrow |\psi_2\rangle = \frac{1}{2\sqrt{2}} \{ (-1)^0 |000\rangle + (-1)^0 |001\rangle + (-1)^0 |010\rangle + (-1)^0 |011\rangle + (-1)^0 |100\rangle + (-1)^0 |101\rangle + (-1)^0 |110\rangle + |(-1)^1 |111\rangle \} \otimes \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

$$= \frac{1}{2\sqrt{2}} \{ |000\rangle + |001\rangle + |010\rangle + |100\rangle + |101\rangle + |110\rangle - |111\rangle \} \otimes \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

Now, we ignore the last qubit from the second register than

$$|\psi_2\rangle = \frac{1}{2\sqrt{2}} \{ |000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle - |111\rangle \}$$

Now for the Deutsch-Jozsa Algo.

$|\psi_3\rangle = H^{\otimes 3}(|\psi_2\rangle)$ is done in step 4. In the final step, we calculate the prob of getting $|0\rangle^{\otimes n}$ ie $\frac{1}{2^6} \left[\sum_{x \in \{0,1\}^3} (-1)^{f(x)} \right]^2$

Hence $f(1,1,1) = 1$ and for all other cases f outputs 0.

Hence the prob. of getting $|0\rangle^{\otimes n}$ is $\frac{1}{2^6} \left[(-1)^0 + (-1)^0 + (-1)^0 + (-1)^0 + (-1)^0 + (-1)^0 + (-1)^0 + (-1)^1 \right]^2$

$$= \frac{1}{2^6} [1+1+1+1+1+1+1-1]^2$$

$$= \frac{6^2}{2^6} = \frac{36}{64} \simeq 0.5625$$

Note that, for Deutsch Jozs Algo, f is used to be either balanced or constant function. The given f outputs 1 in one case and 0 in other.

Hence f is neither constant or Balanced. Therefore executing Deutsch Jozsa on the given f won't provide us any useful information