MADHAV SANKAR KRISHNAKUMAR

40569 2669

1) $N = 4$

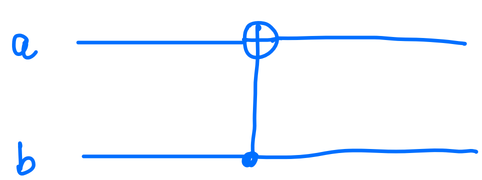
$a = 3$

$$M_a |x\rangle = |ax \pmod N\rangle$$

$$x = \{0, 1, 2, 3\}$$

$$M_a |x\rangle = \{0, 3, 2, 1\}$$

$$\begin{array}{c} \\ 00 \\ 01 \\ 10 \\ 11 \end{array} \begin{array}{c} \begin{array}{cccc} 00 & 01 & 10 & 11 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$



2) $x_0 \wedge x_1$

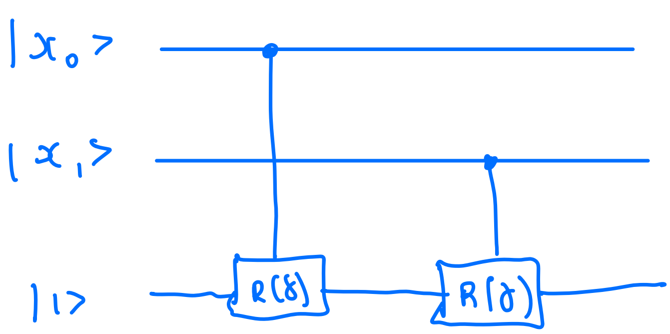
$$S_{ep}(\gamma) = e^{-i\gamma C}$$

x_0, x_1

	count
00	0
01	1
10	1
11	2

$$S_{ep}(\gamma)\left(\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle\right)$$

$$= \frac{1}{2}|00\rangle + \frac{1}{2}\left(e^{-i\gamma}\right)|01\rangle + \frac{1}{2}\left(e^{-i\gamma}\right)|10\rangle + \frac{1}{2}\left(e^{-i\gamma}\right)^2|11\rangle$$

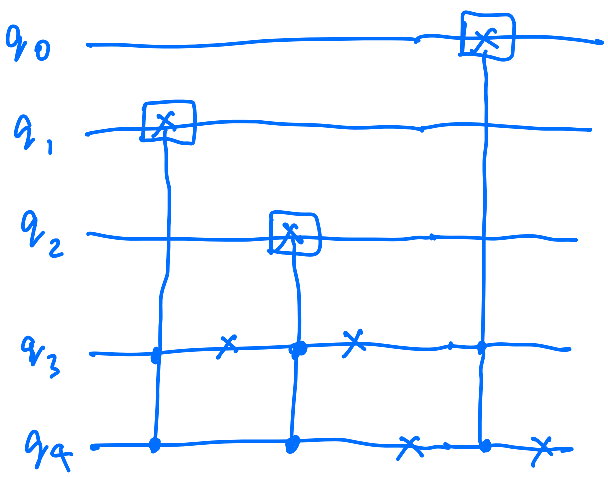


3) 00 → nothing

10 → flip 1^{st}

11 → flip 2^{nd}

01 → flip 3^{rd}




```
OPENQASM 2.0
include "qelib1.inc";
qreg q[5]
creg c[5]
ccx q[3], q[4], q[1]
x q[3]
ccx q[3], q[4], q[2]
```

```
x    q[3]
x    q[4]
ccx  q[3], q[4], q[0]
x    q[4]
```

4) Assume the list of weighted states are of length 2^n → All weights there, even if it is 0.

CODE:

```
def executeCNOT(state, operands):
size = len(state)
newState = [complex(0, 0)] * size
nQubits = int(math.log(size, 2))
control = operands[0]
target = operands[1]

for i in range(size):
    if state[i] != 0:
    binary = decimalToBinary(i, nQubits)

    if binary[control] == '0':
        newState[i] += state[i]
    else:
        if binary[target] == '0':
            binary = setString(binary, target, '1')
        else:
            binary = setString(binary, target, '0')

            decimal = binaryToDecimal(binary)
            newState[decimal] += state[i]

    return newState

def decimalToBinary(self, x, n):
    s = bin(x).replace("0b", "")
    currsize = len(s)
    ans = '0' * (n - currsize)
    ans += s
    return ans

def binaryToDecimal(self, x):
    return int(x, 2)

def setString(self, s, pos, value):
    newString = ""
    for spos in range(len(s)):
        if spos == pos:
            newString += value
        else:
            newString += s[spos]
    return newString
```

5) R_x, R_z, CNOT

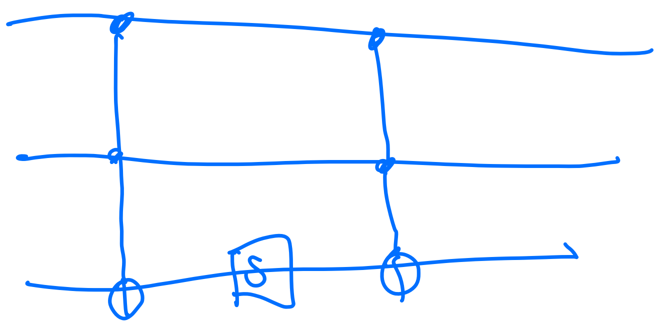
CS

$S = TT$

$T = e^{i\pi/8} \; R_z(\pi/4)$

$i = \cos\theta + i\sin\theta$

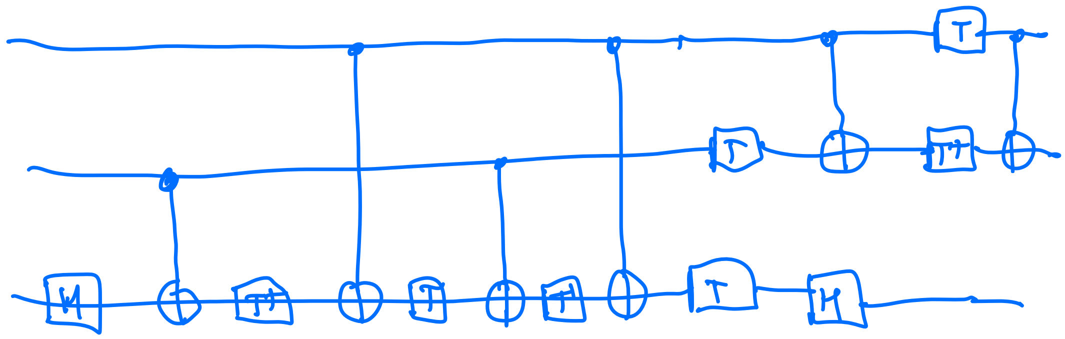
$\theta = \pi/2$



$S = TT$

$T = e^{i\pi/8} \; R_z(\pi/4)$

Implement CCNOT using CNOT



$$T^{\dagger} = e^{-i\pi/8} R_z(-\pi/4)$$

$$T = e^{i\pi/8} R_z(\pi/4)$$

$$H = i R_y(-\pi/2) R_x(\pi)$$

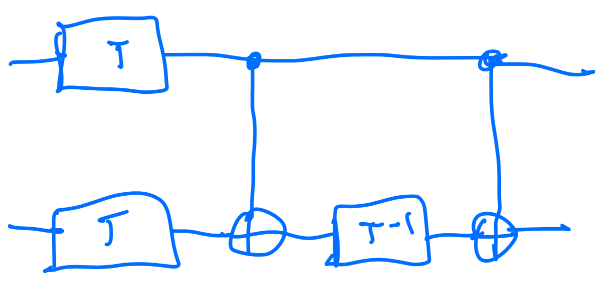
$$R_y(\theta) = R_z(-\pi/2) R_x(\theta) R_z(\pi/2)$$

OR

$$T = \sqrt{S} = Z^{1/4}$$

$$S = Z^{1/2}$$

$$CS = CZ^{1/2}$$



$$T = e^{i\pi/8} R_z(\pi/4)$$

$$T^{-1} = D = -e^{i\pi/8} R_z(-\pi/4)$$

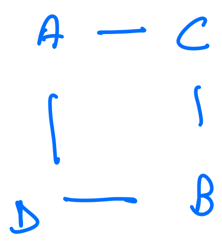
$$CS(i,j) = (T \otimes T)\, CNOT(i,j)\, (I \otimes T^{-1})\, CNOT(i,j)$$

6) $A \to 0$

$B \to 2$

$C \to 3$

$D \to 1$



$CNOT(A,B) = 2$

$CNOT(A,C) = 0$

$CNOT(A,D) = 0$

$CNOT(B,D) = 0$

$CNOT(C,D) = 0$

Totally 2 //

As A is connected to all the other qubits, we need to have one pair of (A, some qubit) to swap, hence 2 is minimum //

7) h q[1]

s q[0]

cx q[0], q[1]

s q[0]

cx q[0], q[1]

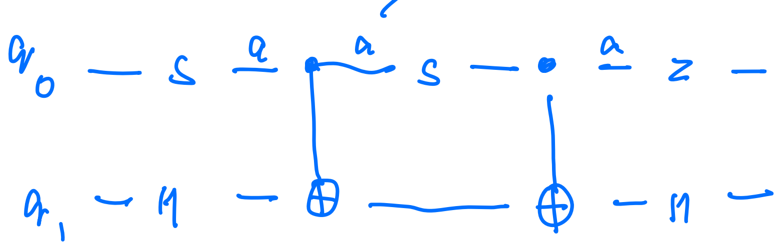
z q[0]

h q[1]

$$S = TT = R_z(2\pi)$$

$$Z = S^2 = R_z(4\pi)$$

only phase shifts



So all 'a'

↓

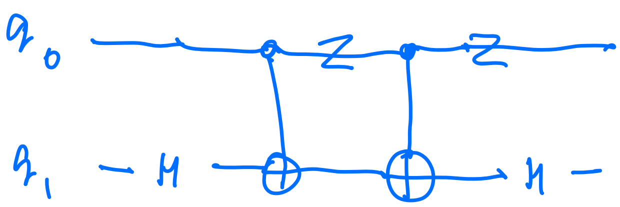
merge the phase shifts

Use "rotation merging"

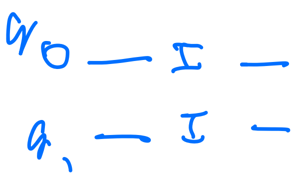
merge the S, S, Z

$Z = S^2$

$ZZ = I$] CANCELLATION



CNOT CNOT = I] CANCELLATION
HH = I



I //

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

8) Phase Estimation

$Z \qquad |1\rangle$

$n = 1$

$\psi_0 \quad \psi_1 \rightarrow 2$ eigenvectors

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

$Av = \lambda v$

$(A - \lambda I)v = 0$

$\det(A - \lambda I) = 0$

$(\lambda - 1)(\lambda + 1) = 0$

$\left.\begin{array}{l} \lambda_1 = 1 \\ \lambda_2 = -1 \end{array}\right\}$ eigenvalues

$(A - \lambda I)v = 0$

$\lambda_1 = 1$

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

$$x_2 = 0$$

$$\begin{bmatrix} x_1 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} 1 \\ 0 \end{bmatrix} //$$

$$\lambda_2 = -1$$

$$\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} v = 0 \qquad x_1 = 0$$

$$\begin{bmatrix} 0 \\ x_2 \end{bmatrix} \qquad \begin{bmatrix} 0 \\ 1 \end{bmatrix} //$$

Eigenvectors $\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Eigenvalues $1, -1$

$$e^{i\theta} = 1$$

$$\cos\theta + i\sin\theta = 1$$

$$\theta = 0 \quad \text{or} \quad 2\pi$$

$$e^{i\theta} = -1$$

$$\theta = \pi$$

We need $e^{2\pi i\theta}$

$$\theta = 0 \text{ or } 1$$

$$\theta = 1/2$$

$$\varepsilon \leq \frac{1}{2^{m+1}}$$

$$m = 1$$

$$\varepsilon \leq \frac{1}{2^2}$$

$$\varepsilon \leq \frac{1}{4}$$

$$P_j > 4/\pi^2 > 0.4$$

It will give an approximation of θ that has first n bit correct.