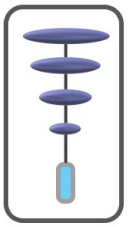


Preâmbulo

O nosso mundo é um sistema mecânico quântico. As partículas fundamentais têm propriedades(carga, massa, spin, posição, etc) que podem ser descritas usando informações quânticas.

A computação quântica é uma tecnologia em pleno desenvolvimento, que promete superar os actuais e mais poderosos computadores do mercado.

Essa capacidade é devida ao facto de um qubit se basear na física quântica e poder funcionar ao mesmo tempo com base em multiplos estados, ao invés dos dois estados (o "zero" *ou* o "um") do bit dos computadores clássicos, tornando possivel a computação dos vários estados simultaneamente em paralelo com apenas um circuito.

Isso permite que um computador quântico possa realizar cálculos em *alguns segundos* para a resolução de problemas altamente complexos, enquanto o maior computador actual poderia necessitar de *muitos milhares de anos* para realizar o mesmo trabalho. Todos os problemas computacionais resolvidos por um computador quântico com poucos qubits (a partir de um certo número de qubits, isso já não é possível) podem igualmente ser resolvidos por um computador clássico, é uma questão do tempo necessário para o respectivo processamento.

Os computadores quânticos podem ser construídos com qualquer sistema de partículas (átomos, protões, fotões, etc.) que satisfaçam as leis da física quântica.

O poder da computação quântica surge, em parte, porque a dimensão do espaço vetorial dos vetores de estado quântico cresce exponencialmente com o número de qubits. O resultado de uma computação quântica é sempre 0s e 1s (“zeros” e “uns”).

O objectivo da construção de computadores quânticos não é que se deseje substituir os computadores clássicos por computadores quânticos, o que é pouco provável, e talvez nunca venha a acontecer, mas sim dispor de poder computacional para resolver problemas complexos.

Para entender minimamente o funcionamento da computação quântica, é conveniente ter presentes algumas noções básicas de mecânica quântica e também, antes desta, de mecânica clássica. Em termos de matemática, são fundamentais noções de álgebra linear, espaços vectoriais complexos, trignometria, números complexos, álgebra booleana, etc.. Em informática, uma ou mais linguagens de programação que suportem o desenvolvimento de programas nesta área. Para programar, é necessária a instalação de um kit de desenvolvimento de software (SDK) com as necessárias bibliotecas, compiladores, emuladores, etc..

Bit

O termo bit, no qual se baseia o sistema binário, resulta da mistura de duas palavras **bi**nary digi**t**, é a menor unidade de informação do computador clássico.

É um sistema construído por dois dígitos diferentes, cujos estados clássicos são 0 e 1, constituindo assim o alfabeto mais simples do mundo e com o qual, só com dois símbolos os computadores clássicos

gravam em memória e processam tudo, incluindo os circuitos quânticos, sendo possível, escrever, armazenar, editar e divulgar bibliotecas infinitas de livros, imagens, vídeos, comunicações, e na construção, assistência, funcionamento e controlo de toda a tecnologia existente, desde uma simples máquina de lavar roupa até aos telemóveis, televisores, elevadores, rádios, relógios, máquinas fotográficas, máquinas de calcular, computadores, drones, robots, carros, aviões, barcos, mísseis, foguetões, satélites, aparelhos de tomografia (TC), sistemas de medidação, conservação, semáforos de trânsito, escritórios, hospitais, centros operacionais, etc., literalmente, a sociedade humana depende directamente em quase tudo desta tecnologia construída com o código binário, com a electrónica e com a sofisticação do firmware e do software.

O conjunto de estados clássicos do bit pode ser considerado como $\sum=\{0,1\}$, em que $\sum$ é um conjunto finito e não vazio.

O sistema binário funciona como o sistema decimal, sendo que este último usa 10 algarismos como base e o sistema binário tem apenas dois algarismos. A associação de n bits resulta num conjunto de 2^n combinações possíveis. Para um conjunto de 8 bits, $n=8$, designado por byte, resultam 256 combinações diferentes, representando cada uma destas os caracteres alfanuméricos, caracteres especiais, símbolos, etc..

Por exemplo, o meu nome escrito no sistema binário:

```
01000011 01100001 01110010 01101100 01101111 01110011 01000011
01101000 01100001 01101101 01100010 01100101 01101100
```

O primeiro destes bytes acima referidos, convertido para o sistema decimal é:

$(0x2^7)+(1x2^6)+(0x2^5)+(0x2^4)+(0x2^3)+(0x2^2)+(1x2^1)+(1x2^0)=67$, que na tabela ascci corresponde ao caracter "C".

Qubit

O Qubit (***Qu**antum **Bit***) é a unidade fundamental de informação na computação quântica. Pode ter um valor 0 ou 1, como no bit clássico, ou uma superposição quântica de ambos, isto é, uma matriz com uma infinidade de combinações de 0 e 1, considerando-se cada uma dessas superposições com o seu próprio estado quântico. A sua implementação física pode ser realizada recorrendo a dispositivos quânticos de dois estados, por exemplo, átomos, iões, à polarização de fotões, spin de electrões, etc, bastando para tal que se encontre dois estados suficientemente isolados dos restantes.

O qubit é um espaço vectorial e o seu estado é apresentado como um vector dado pela matriz:

$$\begin{bmatrix} a \\ b \end{bmatrix}$$

Estados do qubit

Os estados quânticos de um qubit são definidos como vectores em espaços espaciais complexos.

Representação de um estado quântico de dois níveis, na base de dois vectores $|0\rangle$ e $|1\rangle$:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

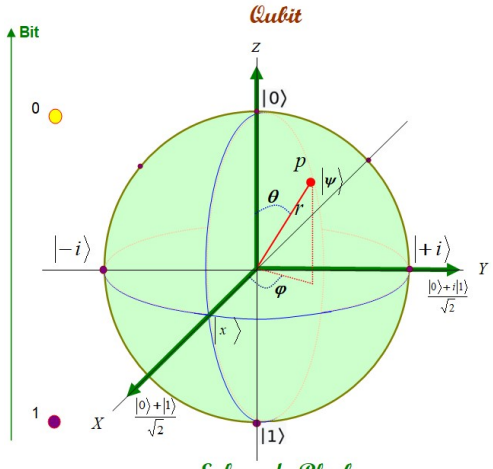


fig. 1 Carlos Chambel/20240422

considerando o sistema de coordenadas esféricas:

- r é a distância entre a origem e o ponto $p(r,\varphi,\theta)$
- φ é o ângulo (em radianos) formado entre o eixo x e a projecção de r no plano xy
- θ é o ângulo (em radianos) formado entre r e o eixo z

Onde: $r \geq 0$; $0 \leq \theta \leq \pi$; $0 \leq \varphi \leq 2\pi$

Então:

$$x = r\sin(\theta)\cos(\varphi)$$

$$y = r\sin(\theta)\sin(\varphi)$$

$$z = r\cos(\theta)$$

e que:

$$r = \sqrt{x^2 + y^2 + z^2}$$

Podendo ser definidos pela combinação linear:

$$|\psi\rangle = a_0|0\rangle + a_1|1\rangle$$

Onde $|\psi\rangle$ define uma superposição dos estados $|0\rangle$ e $|1\rangle$, sendo a_0 e a_1 os coeficientes complexos que representam as amplitudes de probalidades desses estados .

Os elementos a e b da matriz representam a probalidade de o qubit entrar em colapso, sendo $|a|^2$ correspondente à probalidade de o qubit colapsar para zero, e o novo estado do qubit é $\begin{bmatrix}1\\0\end{bmatrix}$ e $|b|^2$ a probalidade de o qubit colapsar para um , e o novo estado do qubit é $\begin{bmatrix}0\\1\end{bmatrix}$.

Considerando que $a,b \in C$ e a soma das probalidades (de qualquer sistema quântico) é igual a 1.

$$|a|^2 + |b|^2 = 1$$

Partindo da fórmula de Euler (ver apontamento na página 17 \\www.carlos-ch-santos.net/deta/Tutorial_FFT.pdf):

$$e^{i\theta} = \cos(\theta) + i\sin(\theta)$$

resulta a forma geral de um estado de um qubit:

$$|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\varphi}\sin(\frac{\theta}{2})|1\rangle$$

Ou, em termos vectoriais:

$$|\psi\rangle = \begin{bmatrix}\cos(\frac{\theta}{2})\\ e^{i\varphi}\sin(\frac{\theta}{2})\end{bmatrix}$$

Para um sistema com n qubits, o estado quântico desse sistema pode ser descrito como:

$$|\psi\rangle = \Sigma_{i=0}^{2^{n-1}} a_i |i\rangle$$

$$\text{Com} \quad a_0, \ldots a_{2^{n-1}} \in C \quad \text{e} \quad \Sigma_{i=0}^{2^{n-1}} |a_i|^2 = 1$$

O número de estados quânticos cresce exponencialmente com o número de qubits. Num computador quântico composto por n qubits podem existir em sobreposição 2^n estados. Por exemplo, num computador quântico composto por 100 qubits, o número de estados quânticos em sobreposição pode ser 2^{100} ao mesmo tempo, o correspondente a 1,26765060022823 x 10^{30} bits.

qubits	correspondência a bits
1	2
2	4
3	8
4	16
16	65 536
17	131 072
18	262 144
32	4 294 967 296
33	8 589 934 592
34	17 179 869 184
64	18 446 744 073 709 600 000
65	36 893 488 147 419 100 000
88	309 485 009 821 345 000 000 000 000
98	316 912 650 057 057 000 000 000 000 000
99	633 825 300 114 115 000 000 000 000 000
100	1 267 650 600 228 230 000 000 000 000 000

Fig. 2 Carlos Chambel 20240412

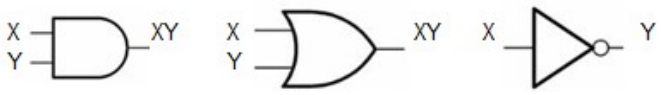
Portas lógicas clássicas

As portas lógicas clássicas são circuitos electrónicos compostos principalmente por transistores, diodos, condensadores e resistências, que aceitam um conjunto de sinais lógicos de entrada para produzirem uma determinada saída de um bit que representa a entrada transformada pelo circuito, implementando operações da álgebra booleana

∧			∨			¬	
X	Y	XY	X	Y	XY	X	Y
0	0	0	0	0	0	0	1
0	1	0	0	1	1		
1	0	0	1	0	1	1	0
1	1	1	1	1	1		

fig. 3 *Carlos Chambel/20240413*

no hardware electrónico



Circuito integrado com 4 portas NAND

Fig.4

Portas lógicas quânticas

As portas lógicas quânticas são operadores lineares unitários sobre um ou mais qubits, transformando o estado dos qubits de entrada (superposição e emaranhamento) em um novo estado. São os blocos de construção dos circuitos quânticos, como as portas lógicas clássicas são para circuitos digitais convencionais, mas, ao contrário destas, são reversíveis e probabilísticas.

Por exemplo, a porta quântica NOT (ou Porta quântica Pauli-X) correspondente à porta clássica NOT, pode ser descrita como:

$$U_{NOT}(|0\rangle) = |1\rangle$$

$$U_{NOT}(|0\rangle) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

e equivale a uma rotação de π radianos no eixo $\mathbf{X}$ da esfera de Bloch, conforme figura 1.

Representação gráfica de uma porta quântica, em que *U* é um rótulo para descrever a função dessa porta:

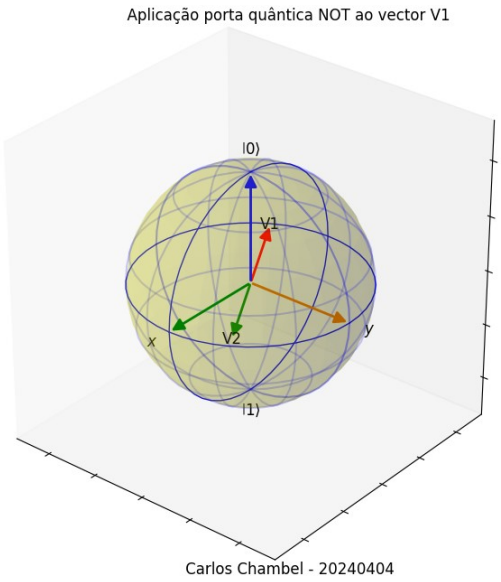


Fig. 5

Algoritmos quânticos

É um processo de representação de uma sequência de instruções lógicas de um determinado problema que pode ser resolvido num computador quântico e que apresente em pelo menos uma das etapa superposição ou emaranhamento. Ao contrário de um algoritmo clássico, o algoritmo quântico é sempre reversível.

Exemplos de algoritmos quânticos: algoritmo de Shor (teoria dos números: fatoração de inteiros); algoritmo de Grover (pesquisa num banco de dados).

Circuitos quânticos

Os circuitos quânticos são blocos de portas quânticas e operadores para a execução de uma determinada tarefa e encontrar a melhor maneira de manipular os qubits (informação) para obter o resultado desejado, executando diversas operações em simultâneo.
São a representação gráfica dos algoritmos quânticos.
Cada linha horizontal representa um registo qubit. A linha superior é o registo do qubit 0 e os restantes são rotulados sequencialmente.
As portas quânticas são representadas em pequenos rectângulos desenhados sobre uma linha do circuito e com a respectiva indicação, por exemplo H = Porta de Hadamard.

O tempo flui da esquerda para a direita.

A porta mais à esquerda é a aplicada em primeiro lugar.

À esquerda, os qubits de entrada e à direita os qubits finais do resultado

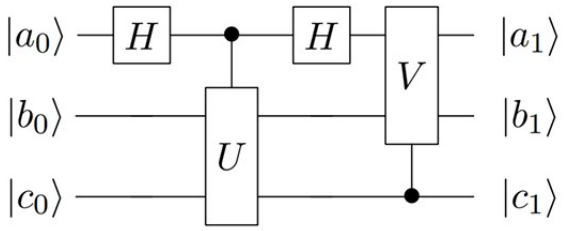


Fig.6

Exemplo de um circuito quântico simples de 4 qubits e output de 4 bits clássicos, com input, computação e output:

```
"""
        Carlos Chambel  20240421
  criando um circuito quântico com 4 qubits
            Python 3.11
"""
# fig.7
from qiskit import QuantumCircuit
qc = QuantumCircuit(4, 4)
qc.draw()
print(qc)
```

Fig.7

Output:

```
q_0:

q_1:

q_2:

q_3:

c: 4/


Process finished with exit code 0
```

Fig.8

Superposição quântica

A superposição quântica é um dos conceitos fundamentais da mecânica quântica, que descreve a capacidade de um sistema quântico estar em múltiplos estados simultaneamente. Isso significa que, em vez de estar em apenas um estado definido, como na física clássica, um objeto quântico pode estar em uma superposição de diferentes estados até que uma medição seja realizada, momento em que colapsa para um estado particular. Em computação quântica significa que os qubits podem representar diversos estados simultaneamente.

A superposição quântica pode ser obtida pela aplicação de uma porta de Hadamard a um circuito de um qubit.

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

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

$$H\left|\Psi_0\right\rangle = \frac{1}{\sqrt{2}}(\left|0\right\rangle + \left|1\right\rangle)$$

Na esfera de Bloch a porta Hadamard é o equivalente a uma rotação no eixo Y de 90º e de seguida a uma rotação no eixo X de 180º .

```
"""
        Carlos Chambel  20240419
    criação de uma superposição de estados num qubit
    aplicação de uma porta quântica Hadamard num
    qubit de um circuito quântico com 3 qubits e 3 bits
              Python 3.11
"""
# fig.9
from qiskit import QuantumCircuit
circuit = QuantumCircuit(3, 3)
circuit.h(1)
circuit.measure([0, 1, 2], [0, 1, 2])
circuit.draw()
print(circuit)
```

Fig.9

Output:

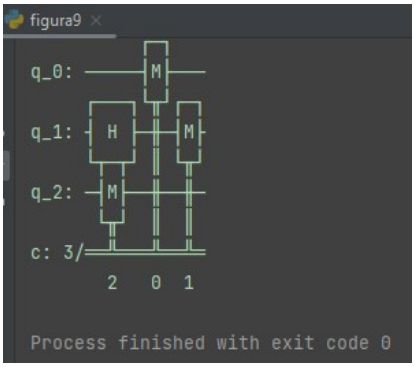


Fig.10

Emaranhamento quântico

O emaranhamento quântico é um efeito que correlaciona o comportamento de duas ou mais partículas separadas, independentemente da distância entre si, seja essa distância um metro ou milhares de quilómetros. Alguns cientistas afirmam que o emaranhamento quântico é um dos temas mais importantes da ciência do século XXI.

Considere-se um sistema *S* descrito pelo vector $|\psi\rangle$ e composto por *A* e *B*. Se não for possível descrever o estado de *S* na forma de produto tensorial $|\psi\rangle = |a\rangle \otimes |b\rangle$ onde $|a\rangle$ e $|b\rangle$ são os estados de *A* e *B*, então *A* e *B* estão emaranhados.

Quando se aproximam dois qubits, de seguida realizada uma operação para o seu emaranhamento e depois separá-los um do outro, o estado quântico de cada um não pode ser descrito independentemente, antes deve ser dado para o sistema como um todo e o que se verifica em um qubit é refletido instantâneamente no outro.

Quando se procede à sua medição esses qubits emaranhados produzirão sempre o mesmo resultado, zero ou um estado perfeitamente aleatório, não importa a distância a que se encontrem um do outro.

Uma vez um qubit emaranhado, nenhum outro em qualquer parte do universo poderá partilhar desse emaranhamento.

Circuito com emaranhamento quântico aplicando uma porta Hadamart no qubit 0, seguida da aplicação de uma porta CNOT entre o qubit 0 e qubit 1:

```
"""
        Carlos Chambel   20240417
  criando um circuito quântico e emaranhamento de qubits
            Python 3.11
"""
# fig.11
from qiskit import QuantumCircuit,  Aer, execute
circuit = QuantumCircuit(3, 3)
circuit.h(0)
circuit.cx(0, 1)
backend = Aer.get_backend('statevector_simulator')
job = execute(circuit, backend)
result = job.result()
statevector = result.get_statevector()
circuit.measure_all()
circuit.draw()
print(circuit)
print(statevector)
```

Fig.11

Output do circuito e do vector de estado:

```
     ┌───┐     ┌─┐
q_0: ┤ H ├──■──┤M├──────
     └───┘┌─┴─┐└╥┘┌─┐
q_1: ─────┤ X ├─╫─┤M├───
          └───┘ ║ └╥┘┌─┐
q_2: ───────────╫──╫─┤M├
                ║  ║ └╥┘
c: 3/═══════════╩══╩══╩═
                0  1  2
meas: 3/
Statevector([0.70710678+0.j, 0.        +0.j, 0.        +0.j,
             0.70710678+0.j, 0.        +0.j, 0.        +0.j,
             0.        +0.j, 0.        +0.j],
            dims=(2, 2, 2))

Process finished with exit code 0
```

Fig.12

Medição estados quânticos

O qubit pode ser 0 ou 1, ou os dois ao mesmo tempo, mas quando um sistema num estado quântico é submetido à observação a medição produz o resultado um bit clássico probabilístico, isto é, um estado clássico.

Operador de medição

O computador quântico pode estar numa superposição de milhões de estados quânticos ao mesmo tempo, mas quando o medimos, obtemos apenas um único estado clássico.

Com a medição de um estado quântico $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ obtém-se um bit clássico probabilistico, o qual é 0 com uma probabilidade de $|a_0|^2$ ou 1 com uma probabilidade de $|a_1|^2$.

Exemplo, para um qubit no estado quântico:

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

resulta nos dois resultados possíveis, "0" e "1" com as seguintes probabilidades:

para o resultado "0" = $\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$

para o resultado "1" = $\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$

dado que:

$$|a|^2 + |b|^2 = 1$$

Isto é, o resultado será "0" em 50% e "1" em 50% das vezes no estado quântico que é medido.

Considerando um sistema constituído por dois qubits, o estado resultado do produto tensorial dos dois qubits:

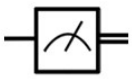
$$|\psi\rangle = (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle)$$

$$= a_1 a_2 |00\rangle + a_1 b_2 |01\rangle + b_1 a_2 |10\rangle + b_1 b_2 |11\rangle$$
$$= a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle$$

dado que:

$$|a_{00}|^2 + |a_{01}|^2 + |a_{10}|^2 + |a_{11}|^2 = 1$$

Nos circuitos quânticos, o operador de medição é apresentado pelo grafismo. À direita, o traço o traço representa um qubit e à esquerda, o traço duplo representa um bit clássico:



Exemplo de um circuito simples, com medição dos qubits

```
"""
        Carlos Chambel  20240411
  criando um circuito quântico com 3 qubits
            Python 3.11
"""
# fig.13
from qiskit import QuantumCircuit
circuit = QuantumCircuit(3, 3)
circuit.measure([0, 1, 2], [0, 1, 2])
circuit.draw()
print(circuit)
```

Fig.13

e respectivo resultado:

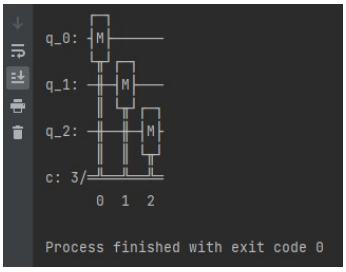


Fig.14

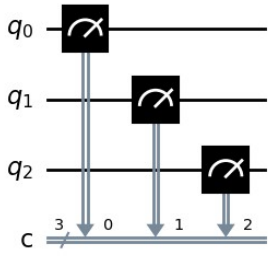


Fig.15

Interferência quântica

A probabilidade do qubit entrar em colapso de uma forma ou de outra é determinada pela interferência quântica. A interferência quântica afeta o estado de um qubit para influenciar a probabilidade de um determinado resultado durante a medição, e esse estado probabilístico é onde o poder da computação quântica se sobressai.

Função de onda quântica

Um estado quântico pode ser descrito, num determinado momento, por um vector de estado ou por uma função complexa de momento e posição, a função onda. Quando temos muitos qubtis emaranhados, todas as suas funções de onda são somadas em uma única função de onda geral que descreve o estado o computador quântico.

Essa função onda pode resultar de forma construtiva formando uma onda maior, interferir destrutivamente cancelando-se. Esta função de onda define as diferentes probalidade dos diferentes estados dos qubits. Alterar os estados de diferentes qubits pode resultar na alteração das probalidades de diferentes estados quando se mede o computador quântico.

Uma interferência construtiva aumenta a probalidade de uma resposta correcta e uma interferencia destrutiva usa-se para diminuir as probalidade de respostas incorrectas.

Simulação de circuitos quânticos num computador clássico

Todos os problemas computacionais resolvidos por um computador quântico (com poucos qubits) podem igualmente ser resolvidos por um computador clássico, é uma questão do tempo necessário para o respectivo processamento.

Exemplo de um programa a processar num computador clássico, criando um circuito quântico, processar e obter os mesmos resultados como se processado num computador quântico.

Quando se realiza uma medida de um estado quântico em superposição, a probabilidade é encontrar o estado numa das configurações possíveis (0 ou 1), já que não é possível medir o estado em simultâneo. Mas executando essa medida 1000 vezes, por exemplo, os valores podem não ser iguais, isto é, 500 vezes "0" e 500 vezes "1", resultando assim probabilidades diferentes.

```
"""
# carlos chambel  20230831  20240416
#
#   criação circuito quântico
#   qubits em superposição
#   simulação quantica
#   resultado em bits classicos da frequencia por estados quanticos
"""

# figura 16
font1 = {...}
font2 = {...}
import ...
sho = 10000
qc = QuantumCircuit(3, 2)
qc.h(0)
qc.h(1)
qc.h(2)
qc.measure([0, 1], [0, 1])
backend = BasicAer.get_backend('qasm_simulator')
job = execute(qc, backend, shots=sho)
resulta = job.result()
counts = resulta.get_counts()
```

Fig.16

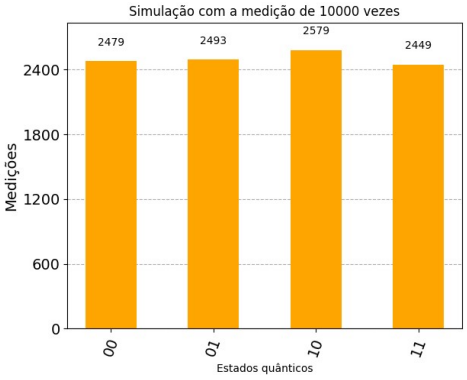


FIG.17 Carlos Chambel/20240429

Processamento de circuitos quânticos
num computador quântico

Algumas das grandes empresas desta área disponibilizam o acesso directo a computadores quânticos com mais de 100 qubits, para submeter a processamento os circuitos quânticos criados pelo utilizador, e consequentemente, a obtenção dos respectivos outputs. É necessária a criação de uma conta no site, e no respectivo dashboard é possível acompanhar o estado da queue onde se encontra o processo, tempos gastos, etc..

- - - x - - -