

## Brazil Quantum Camp

### Quantum Tech - Lista de Exercícios 1

Nome Completo:

CPF:

#### Orientações:

- Por favor, digite suas respostas diretamente nos campos editáveis fornecidos.
- Certifique-se de salvar o PDF após completar todos os exercícios.
- O PDF salvo com as respostas deverá ser enviado conforme as instruções fornecidas na plataforma.

#### Questão 1

O que acontece com um qubit que está em estado de superposição no momento em que uma medição é realizada?

- A) Ele permanece em superposição para permitir cálculos paralelos contínuos.
- B) Ocorre o colapso da função de onda e o qubit assume apenas um dos estados possíveis.
- C) O qubit é instantaneamente clonado para permitir a medição.
- D) Sua energia é alterada conforme a equação de Planck.

#### Questão 2

Considere dois qubits inicialmente no estado  $|\psi_0\rangle = |0\rangle\otimes|1\rangle \in \mathbb{C}^2 \otimes \mathbb{C}^2$ . Aplica-se primeiro a porta de Hadamard H apenas no primeiro qubit e, em seguida, a porta CNOT, na qual o primeiro qubit é o controle e o segundo é o alvo. Sobre o estado final do sistema, assinale a alternativa correta.

- A) O estado final é separável e pode ser escrito como produto tensorial de dois estados de um qubit.
- B) O estado final é um estado de Bell e apresenta emaranhamento máximo.
- C) O estado final pertence a  $\mathbb{C}^2$ , pois a CNOT reduz a dimensionalidade do espaço de Hilbert.
- D) O estado final é um autovetor da porta CNOT associado a um autovalor complexo.

### Questão 3

Complete a função:

```
1 def eh_porta(porta: np.ndarray) -> bool:  
2     ...
```

Essa função deve identificar se o parâmetro `porta` é uma matriz que representa uma porta lógica quântica. Considere que `porta` é um array do NumPy que representa uma matriz.

A função `eh_porta` deve retornar:

- `True`, se a matriz representar uma porta lógica quântica;
- `False`, caso contrário.

Considere que o seguinte import já foi realizado e nenhum outro import pode ser adicionado. O código não será executado caso haja qualquer instrução de import adicional.

```
1 import numpy as np
```

Implemente a função `eh_porta` no campo abaixo, respeitando todas as restrições descritas acima.

#### Questão 4

No protocolo de codificação superdensa, Alice e Bob compartilham inicialmente o estado de Bell

$$|\Phi^+\rangle_{AB} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Para transmitir dois bits clássicos a Bob, Alice aplica operações locais apenas no seu qubit A e, em seguida, envia esse qubit a Bob. Bob então aplica uma porta CNOT (controle em  $A$ , alvo em  $B$ ), seguida de uma Hadamard no qubit  $A$ , e mede ambos os qubits na base computacional.

Sabe-se que Alice aplicou a sequência de portas Z seguida de X sobre o seu qubit antes de enviá-lo a Bob.

Com base no circuito e nas operações realizadas, quais bits clássicos Alice quis transmitir?

- A) 00
- B) 01
- C) 10
- D) 11

#### Questão 5

Você deve implementar a função de decodificação realizada por Bob no protocolo de teletransporte quântico. Essa função representa a etapa final do protocolo, na qual Bob utiliza as informações clássicas recebidas de Alice para aplicar a correção adequada ao seu qubit.

A função recebe três parâmetros:

- `bell`: um número inteiro que indica qual par de Bell foi utilizado como estado emaranhado entre Alice e Bob
- `b1`: o bit clássico obtido da medição do qubit da mensagem
- `b2`: o bit clássico obtido da medição do qubit emaranhado de Alice

No protocolo original de teletransporte quântico, o par emaranhado utilizado é o estado:  $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Nesta implementação, diferentes pares de Bell podem ser utilizados, conforme a codificação abaixo:

- 0 representa o estado  $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$
- 1 representa o estado  $\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$
- 2 representa o estado  $\frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$
- 3 representa o estado  $\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$

A função a ser implementada é:

```
1 def decodificar(bell: int, b1: bool, b2: bool) -> np.array  
2     ...
```

A função deve determinar, com base no valor de `bell` e nos bits clássicos `b1` e `b2`, qual operação quântica Bob deve aplicar ao seu qubit para corrigir o estado recebido e recuperar o estado original da mensagem.

A operação retornada deve ser uma matriz unitária  $2 \times 2$  do tipo `np.ndarray`. Essa matriz pode ser a identidade, a porta de Pauli X, a porta de Pauli Z ou a combinação das portas X e Z.

Considere que o seguinte import já foi realizado e nenhum outro import pode ser adicionado. O código não será executado caso haja qualquer instrução de import adicional.

```
1 import numpy as np
```

## Questão 6

Considere dois observadores, Alice e Bob, que compartilham o estado emaranhado de dois qubits  $|\Psi^-\rangle_{AB} = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$ , conhecido como estado singlet.

Em teorias clássicas baseadas em variáveis ocultas locais, as correlações entre medidas devem satisfazer a desigualdade de Bell na forma CHSH:

$$|E(a_0, b_0) - E(a_0, b_1)| + |E(a_1, b_0) + E(a_1, b_1)| \leq 2$$

Como apresentado nos slides, Alice mede os observáveis

$$A_0 = Z^A, \quad A_1 = X^A$$

enquanto Bob mede

$$B_0 = -\frac{Z^B + X^B}{\sqrt{2}}, \quad B_1 = \frac{Z^B - X^B}{\sqrt{2}}.$$

Nas quais  $X$  e  $Z$  são as matrizes de Pauli.

Sabendo que, para o estado singlet,  $\langle Z^A Z^B \rangle = -1$ ,  $\langle X^A X^B \rangle = -1$ ,  $\langle Z^A X^B \rangle = 0$  e  $\langle X^A Z^B \rangle = 0$ , calcule o segundo termo da desigualdade,  
 $E(a_0, b_1) = \left\langle Z^A \otimes \frac{Z^B - X^B}{\sqrt{2}} \right\rangle$ .

Para sua resposta, use uma precisão de 7 dígitos após a vírgula.

Resposta:

## Questão 7

Seu desafio é identificar qual é a porta quântica secreta aplicada a um qubit.

Complete a seguinte função:

```
1 def qual_porta(porta) -> int:  
2     ...
```

O parâmetro `porta` é uma função que aplica uma porta lógica quântica desconhecida, que pode ser I, X, Y ou Z. Essa função só pode ser chamada uma única vez.

Para aplicar a porta, utilize obrigatoriamente a chamada:

```
1 porta(qc, qubit)
```

onde `qc` é um objeto do tipo `QuantumCircuit` do Qiskit e `qubit` é um inteiro que indica o índice do qubit ao qual a porta será aplicada.

A função `qual_porta` deve identificar qual porta foi aplicada e retornar:

- 0 se a porta for I
- 1 se a porta for X
- 2 se a porta for Y
- 3 se a porta for Z

Considere que os seguintes imports já foram realizados e nenhum outro import pode ser adicionado. O código não será executado caso haja qualquer instrução de import adicional.

```
1 from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit, generate_preset_pass_manager  
2 from qiskit_ibm_runtime import Sampler  
3 from qiskit_aer import AerSimulator
```

Implemente a função `qual_porta` no campo abaixo, respeitando todas as restrições descritas acima.

### **Questão 8**

Quais dos algoritmos quânticos abaixo são determinísticos, ou seja, requerem apenas uma execução para resolver o problema?

- A) Deutsch
- B) Deutsch-Josza.
- C) Bernstein-Vazirani..
- D) Todos são determinísticos.