

動かしながら始める 量子コンピュータ

最終回

第4回 量子プログラミング・コンテストの問題に挑戦

束野 仁政

量子コンピュータが、どのような原理で動作し、どのような問題を解決し、どのように扱えるのかを紹介します。最終回となる今回は、過去に行われたコンテストの問題に挑戦してみます。

ここでとりあげる量子プログラミング・コンテストの問題

2018年7月6日～9日に、Microsoft Q# Coding Contest-Summer 2018 (<https://codeforces.com/contest/1002>) という量子プログラミング・コンテストが開催されました。これは、マイクロソフトが開発しているQ#という量子プログラミング言語を利用して、量子コンピュータに関する15個の問題を解く時間を競うコンテストです。

コンテストに出題された問題から2問を取り上げ、Qiskitでの解答を紹介します。本稿で紹介する解答は、数ある解き方の1つと考えてください。

解法に関する公式の解説は次を参照してください。

<https://assets.codeforces.com/rounds/997-998/main-contest-editorial.pdf>

また、マイクロソフトは2019年3月にも量子プログラミング・コンテストを開催しました。

<https://codeforces.com/contest/1116>

今後も開催されると思うので、興味のある方はチャレンジしてみてください。

問題 1

初期状態 $|0\dots 0\rangle$ から、全ての量子ビットの重ね合わせを生成せよ。

● 具体例で考える

N 量子ビットの場合、 2^N 個の量子状態の重ね合わせ

になります。例えば、1～3量子ビットの場合は次のようになります。

- 1量子ビットの場合： $\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$
- 2量子ビットの場合： $\frac{1}{2}(|00\rangle+|01\rangle+|10\rangle+|11\rangle)$
- 3量子ビットの場合： $\frac{1}{\sqrt{8}}(|000\rangle+|001\rangle+|010\rangle+|011\rangle+|100\rangle+|101\rangle+|110\rangle+|111\rangle)$

1量子ビットの場合は、第1回で説明したアダマール演算を適用するだけです。

$$|0\rangle \xrightarrow{H} \frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$$

また、2量子ビットの場合、1量子ビット目にあダマール演算を適用し、次に2量子ビット目にあダマール演算を適用すれば、答えにたどり着きます。

$$|00\rangle \xrightarrow[1\text{量子ビット目にあダマール演算を適用}]{H} \frac{1}{\sqrt{2}}|0\rangle(|0\rangle+|1\rangle)$$

リスト1 問題1の答え

1～ N 量子ビット目に順にあダマール演算を適用

```
from qiskit import QuantumCircuit, QuantumRegister,
                    BasicAer, execute

# 問題の解を生成する関数
def solve(circuit, qr):
    for index in range(len(qr)):
        circuit.h(qr[index])

# 量子回路を実行する
qr = QuantumRegister(3, 'x')
circuit = QuantumCircuit(qr)
solve(circuit, qr)
# 量子状態を確認するためのシミュレータ
backend = BasicAer.get_backend('statevector_simulator')
# statevector_simulatorを利用するときはshots=1にする
job = execute(circuit, backend, shots=1)
result = job.result()
print(result.get_statevector(circuit))
```