3. Deutsch-Jozsa Algorithm

以下函數dj\_problem\_oracle以gate的形式返回 n = 4 的 Deutsch-Jozsa oracle， Gate採用 5 個量子位元作為輸入，其中最後量子位元 q[4] 是輸出量子位元， 請以此函式為oracle完成Deutsch-Jozsa Algorithm，並以Aer\_simulater輸出測量到的最高值 。

def dj\_problem\_oracle(problem):

    """Returns a 5-qubit Deutsch-Joza Oracle"""

    qc = QuantumCircuit(5)

    int(problem)

    if problem == 1:

        for q in range(4):

            qc.cx(q, 4)

    elif problem == 2:

        qc.cx(0, 4)

        qc.cx(0, 4)

    elif problem == 3:

        qc.cx(2,4)

    else:

        print("There are only currently 3 oracles in this problem set, returning empty (balanced) gate")

    return qc.to\_gate()

**input**

第一行n為測資筆數。

接下來的n行，每行為problem的值。

**output**

輸出以dj\_problem\_oracle(problem) 作為oracle完成Deutsch-Jozsa Algorithm，並以Aer\_simulater輸出測量到的最高值 。

**Sample input**

2

1

2

**Sample output(以下值不是正確解，但須照以下格式輸出)**

0000

0000