

In [1]:

```
import numpy as np

# Importing standard Qiskit libraries
from qiskit import QuantumCircuit, transpile, Aer, IBMQ
from qiskit.tools.jupyter import *
from qiskit.visualization import *
from ibm_quantum_widgets import *
from qiskit.providers.aer import QasmSimulator

# Loading your IBM Quantum account(s)
provider = IBMQ.load_account()
```

In [2]:

```
from qiskit import *

q=QuantumRegister(2,'q')

M=ClassicalRegister(1,'c')
#f(x1,x2,x3) = x1x2+x2x3+x3x1
#we are trying to check whether f is balanced or constant using Duetsch-Jozsa algo
rithm
DJf = QuantumCircuit(q,M)

#Now we are making circuit
DJf.x(q[1])
DJf.h(q)

#This is quantum equivalent of boolean function f =0
#DJf.cx(q[0],q[1])
#DJf.cx(q[0],q[1])

DJf.h(q[0])
DJf.measure(q[0],M[0])

DJf.draw()

backend = Aer.get_backend('qasm_simulator')
qjob = execute(DJf, backend, shots = 2000)

counts = qjob.result().get_counts()
print(counts)

{'0': 2000}
```

In [ ]: