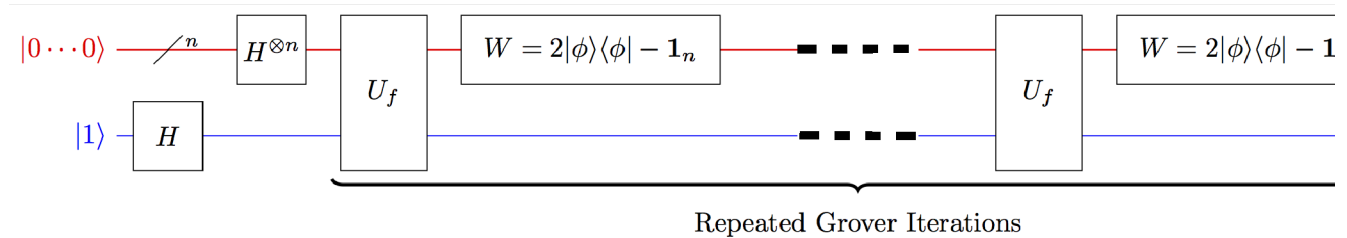


Implementing Grover's Algorithm

We diagram the Grover algorithm



According to the above diagram

$$H^{\otimes n}|0\rangle_n = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} |x\rangle_n \equiv |\phi\rangle_n$$

Also,

$$U_f H^{\otimes n}|0\rangle_n \otimes H|1\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle_n \otimes H|1\rangle = V|\phi\rangle_n \otimes H|1\rangle$$

Next we apply the Grover diffusion operator

$$W = H^{\otimes n}(2|0\rangle\langle 0| - \mathbf{1}_n)H^{\otimes n} = 2|\phi\rangle\langle\phi| - \mathbf{1}_n$$

We repeat this series of gates the integer value of $\sqrt{\pi 2^n/4}$ times prior to making a measurement.
 Lets translate this procedure into code for $n = 7$ qubit.

```

In[1]:= (* we construct the 7-qubit input register *)
zerobit = {1, 0};
onebit = {0, 1};
(* nullket represents the upper register input *)
nullket = Flatten[
    KroneckerProduct[zerobit, zerobit, zerobit, zerobit, zerobit, zerobit, zerobit]];

Hadamard = 1 / Sqrt[2] {{1, 1}, {1, -1}};

(* apply the  $H^{\otimes 7}$  gate on the upper register *)
Hadtimesn = KroneckerProduct[Hadamard,
    Hadamard, Hadamard, Hadamard, Hadamard, Hadamard, Hadamard];

MatrixForm[Hadtimesn];

In[7]:= (* phi represent state  $|\phi\rangle$  defined above *)
phi = Hadtimesn.nullket;

In[8]:= (* tagnumber q is an arbitrary number  $0 < q < 127$  *)
tagnumber = RandomChoice[Table[i, {i, 0, 2^7 - 1}]];

In[9]:= (* construct the oracle for this tagnumber *)
V = DiagonalMatrix[Table[If[i == tagnumber, -1, 1], {i, 0, 2^7 - 1}]];

In[10]:= (* construct gate W *)

W = 2 KroneckerProduct[phi, phi] - IdentityMatrix[2^7];

MatrixForm[W];

(* Grover operator  $G = W V$  *)

In[12]:= (* Grover operator *)
G = W.V;

In[13]:= (* Construct the first six Grover iteration, grover1,
    grover2 ...grover 6 is the output register to be measured *)

grover1 = N[G.phi];
grover2 = N[G.grover1];
grover3 = N[G.grover2];
grover4 = N[G.grover3];
grover5 = N[G.grover4];
grover6 = N[G.grover5];

```

Measurements

```
In[19]:= binaryoutput = Table[IntegerDigits[i, 2, 7], {i, 0, 2^7 - 1}];
```

We assume that six Grover iterations have been performed and the final state of the upper register $|\psi\rangle$ is given by *grover6*. Using the Born rule, the probability to measure state $|k\rangle$ is $|\langle k|\psi\rangle|^2$

```
In[20]:= measurement := RandomChoice[grover6^2 -> binaryoutput];
```

```
In[21]:= measurement
FromDigits[%, 2]
```

```
Out[21]:= {1, 1, 0, 0, 0, 1, 1}
```

```
Out[22]:= 99
```

Actual tag number for item is

```
In[23]:= tagnumber
```

```
Out[23]:= 99
```

Exercises

(1) Run the code with only one Grover iteration. Tabulate the number of repeated measurements necessary to arrive at the correct tag number. Repeat for two Grover iterations. How do these results compare with the results obtained above, using 6 Grover iterations.

(2) Construct codes with 7,8,9,10 Grover iterations. Which of these gives the optimal success rate.