FYS3110 – Home Exam

Candidate Number 15229

October 13, 2017

Problem 1

1.1

$$\sigma_x |0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \cdot 0 + 1 \cdot 1 \\ 1 \cdot 0 + 0 \cdot 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |1\rangle$$

$$\sigma_x |1\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \cdot 0 + 0 \cdot 1 \\ 1 \cdot 1 + 0 \cdot 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |0\rangle$$

We see that the Pauli x-spin matrix σ_x acting on a spin in z-direction, flips the spin to the oposite value.

1.2

Our requirement that both $|o\rangle = G|i\rangle$ and $|i\rangle = G|o\rangle$ holds, means that

$$|o\rangle = G|i\rangle = GG|o\rangle$$

Since $|o\rangle = GG|o\rangle$, GG must act as the identity operator, giving that G is unitary:

$$GG = I \quad \rightarrow \quad G = G^{-1}$$

If G were a hermitian operator, the following would hold:

$$\langle i| G |o\rangle = \langle o| G |i\rangle$$

Inserting for $G|o\rangle = |i\rangle$ and $G|i\rangle = |o\rangle$, we get that

$$\langle i|i\rangle = \langle o|o\rangle$$

Since G is meant to "preserve normalization", it is natural to asume that $|o\rangle$ and $|i\rangle$ are normalized to 1, and the statement above holds. Since it holds, G must be hermitian.

1.3

We have defined our qubit basis states to be the z spin-up and spin-down states. We have in exercise 1.1 observed that the Pauli x-spin matrix switches the z-spin basis states. We can therefore represent a NOT gate through a σ_x operator.

If σ_x is hermitian, it should hold that $\sigma_x^\dagger = \sigma_x$. We see that

$$\sigma_x^\dagger = (\sigma_x^T)^* = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \sigma_x$$

If σ_x is unitary, it should hold that $\sigma_x^2 = I$:

$$\sigma_x^2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$$

We can see that σ_x is both unitary and hermitian.

1.4

I choose to refer to the gate operator as H_g, to reserve H for the hamiltonian operator, for later usage.

If H_g is unitary, it should hold that $H_g^2 = I$:

$$H_g^2 = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\cdot 1 + 1\cdot 1 & 1\cdot 1 + 1\cdot(-1)\\ 1\cdot 1 + (-1)\cdot 1 & 1\cdot 1 + (-1)\cdot(-1)\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} = I$$

If H_g is hermitian, it should hold that $H_g^\dagger = H_g$:

$$H_g^\dagger = \frac{1}{\sqrt{2}}\left(\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}^T\right)^* = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}^* = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = H_g$$

We see that H_g is both hermitian and unitary.

Applying H_g upon the basis states gets us

$$H_g\left|0\right\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}0\\ 1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ -1\end{pmatrix} = \frac{1}{\sqrt{2}}(\left|1\right\rangle - \left|0\right\rangle)$$

$$H_g\left|1\right\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ 1\end{pmatrix} = \frac{1}{\sqrt{2}}(\left|1\right\rangle + \left|0\right\rangle)$$

We see that applying H_g to the basis states gives us a superposition of the two states. We recongize these superpositions as the up and down spins in the x-direction. The H_g gate turns the z up-spin into an x up-spin, and the z down-spin into an x down-spin.

1.5

We will implement the H-gate as a magnetic field, $\vec{B}$, turned on for a duration t_1. The time-evolution of a 1/2-spin state (here in the spin-z basis) is governed by

$$\chi_z(t) = \alpha e^{-iE_+ t/\hbar}\left|+\right\rangle + \beta e^{-E_- t/\hbar}\left|-\right\rangle \tag{1}$$

Here, $\left|+\right\rangle$ and $\left|-\right\rangle$ are the eigenstates of the 1/2-spin systems Hamiltonian in the spin-z basis, and E_+ and E_- are their respective eigenvalues(the observed energies of the eigenstates).

The Hamiltonian, H, of a 1/2-spin particle in a magnetic field $\vec{B}$ is[1]

$$H = -\gamma\vec{B}\cdot\vec{S}$$

where $\vec{S} = [S_x, S_y, S_z] = \frac{\hbar}{2}[\sigma_x, \sigma_y, \sigma_z]$, where σ_i are the Pauli spin matrices. This gives the Hamiltonian

$$H = -\gamma\frac{\hbar}{2}\ [B_x\sigma_x,\ B_y\sigma_y,\ B_z\sigma_z]$$

If we wish this to be an implementation of the H_g-gate, they must share a common set of eigenstates. To achive this, we wish to write H as constant times the H_g operator. Since σ_y is the only Pauli spin-matrix containing complex terms, we remove it by setting $B_y = 0$. Further, we see that

$$H_g = \begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} = \frac{1}{\sqrt{2}}(\sigma_x + \sigma_z)$$

This means that the magnetic fields in x and z direction must be of equal strength, and we require that

$$B_z = B_x = B_0$$

[1] I will refer to the Hamiltonian as H, and the gate-operator as H_g.

Since the strength of the magnetic field should equal h, we get that

$$|\vec{B}| = h = \sqrt{B_0^2 + B_0^2} \quad \rightarrow \quad B_0 = \frac{h}{\sqrt{2}}$$

We have now chosen a magnetic field in x-z direction, with equal stregth in each of these directions.

By forcing the Hamiltonian to be written in terms of H_g, we get

$$H = -\gamma\frac{\hbar h}{2}H_g$$

To solve for time-dependency, we need the Hamiltonians eigenstates and -values. This is left to Wolfram Alpha, and results in the states and energies

$$|+\rangle = \begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} \qquad E_+ = \gamma\frac{\hbar h}{2}$$
$$|-\rangle = \begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix} \qquad E_- = -\gamma\frac{\hbar h}{2}$$

Inserting this into equation 1 gives

$$\chi_z(t) = \alpha e^{it'}\begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} + \beta e^{-it'}\begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix}$$

where we have introduced $t' = \gamma\frac{h}{2}t$

We will now solve the duration of which the field must be turned on. This is done for both the spin-up and spin-down state, seperately.

The up-state

Knowing that the z-spin is purely upwards at $t = 0$ gives us the criteria

$$\chi_z(0) = \alpha\begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} + \beta\begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

This is two equations with two unknowns, and is left to Wolfram Alpha. It requires that

$$\alpha = -\frac{1}{2\sqrt{2}} \qquad \beta = \frac{1}{2\sqrt{2}}$$

This gives us the time-expansion

$$\chi_z(t) = -\frac{1}{2\sqrt{2}}e^{it'}\begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} + \frac{1}{2\sqrt{2}}e^{-it'}\begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix}$$

We require that this, at some point $t' = t_1'$, becomes the x-spin-up state, represented as $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ in the spin-z basis. This results in two equation, and we pick out the "second" one (second components in the vector equation). The equations also carry a global phase-factor, which we will just add at the end, when we need it.

$$-\frac{1}{2\sqrt{2}}e^{it_1'} + \frac{1}{2\sqrt{2}}e^{-it_1'} = \frac{1}{\sqrt{2}}$$

Using Eulers formula and multiplying with $2\sqrt{2}$ gives

$$-\cos(t_1') - i\sin(t_1') = \cos(-t_1') + i\sin(-t_1') = 2$$

Since we know that $\cos(-x) = \cos(x)$ and $\sin(-x) = -\sin(x)$, we get

$$-i\sin(t_1') = 1$$

Since we would very much like to have a non-complex time, we require that our global phase factor $e^{i\theta} = \pm i$, as any other choice would introduce a complex component. Since we would also like our time to be positive, we pick the sign on the phase factor such that

$$\begin{aligned} -i\sin(t_1') = 1 \cdot e^{i\theta} &= -i \\ \sin(t_1') &= 1 \\ t_1' &= 2\pi n + \frac{\pi}{2} \end{aligned}$$

for $n \in 0, 1, 2, 3....$
Inserting for t_1' and considering that we would probably like the the duration of the magnetic field to be as short as possible (i.e. $n = 0$), and Inserting for $\gamma = g\frac{\mu_B}{\hbar} = \frac{ge}{2m}$, we get the duration

$$t_1 = \frac{2t_1'}{h\gamma} = \frac{\pi}{h\gamma} = \frac{2\pi m}{hge}$$

The spin-down state

We will now, hopefully, solve for the same time in the spin-down state.

Knowing that the z-spin is purely downwards at $t = 0$ gives us the criteria

$$\chi_z(0) = \alpha \begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} + \beta \begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

This is, again, left to Wolfram Alpha, giving the coefficients

$$\alpha = \frac{\sqrt{2}+1}{2\sqrt{2}} \qquad \beta = \frac{\sqrt{2}-1}{2\sqrt{2}}$$

This gives us the time-expansion

$$\chi_z(t) = \frac{\sqrt{2}+1}{2\sqrt{2}} e^{it'} \begin{pmatrix} 1-\sqrt{2} \\ 1 \end{pmatrix} + \frac{\sqrt{2}-1}{2\sqrt{2}} e^{-it'} \begin{pmatrix} 1+\sqrt{2} \\ 1 \end{pmatrix}$$

We require that this, at some point $t' = t_1'$, becomes the x-spin-down state, represented as $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$ in the spin-z basis. This results in two equation, and we pick out with the "first" one. The phase factor will again be considered at the end.

$$\frac{\sqrt{2}+1}{2\sqrt{2}} e^{it_1'}(1-\sqrt{2}) + \frac{\sqrt{2}-1}{2\sqrt{2}} e^{-it_1'}(1+\sqrt{2}) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Multiplying by $2\sqrt{2}$, writing out all the terms, and using Eulers formula, gives

$$-\cos(t_1') - i\sin(t_1') = \cos(-t_1') + i\sin(-t_1') = 2$$

We see that this is exactly the same we got in the spin-up state. Using the same approach, we will obviously arrive at the same time for the spin-down as we did for the spin-up. This is a great sign that our solution is probably correct. It's also rather convenient to have the $\vec{B}$-field on for the same duration, regardless of wether we wish to flip a spin-up or spin-down particle. Anything else would be somewhat impractical to implement.

To sum up, our implementation of the H_g-gate is a $\vec{B} = \frac{h}{\sqrt{2}}\vec{i_x} + \frac{h}{\sqrt{2}}\vec{i_z}$ field, turned on for a duration $t_1 = \frac{2\pi m}{hge}$.

Problem 2

2.1

We have a set of N integers, and apply an operator f, which will return True at a specific integer i^*. We will call n the number of times we had to apply f in order to figure out i^*, and it ranges anywhere between 1 and

N.[2] The probability that we will find i^* after n applications of f, is uniformly equal to $1/N$:[3]

$$P(n) = \frac{1}{N}$$

The expectation value of the number of guesses needed to find i^* is then given as

$$\langle n \rangle = \sum_{n=1}^{N} n\ P(n) = \frac{1}{N}\sum_{n=1}^{N} n = \frac{1}{N}\left(\frac{N(N+1)}{2}\right) = \frac{N+1}{2}$$

2.2

Applying the operator $F = I - 2\left|i^*\right\rangle\left\langle i^*\right|$ onto a state $\left|i\right\rangle$ gives.

$$F\left|i\right\rangle = I\left|i\right\rangle - 2\left|i^*\right\rangle\left\langle i^*|i\right\rangle$$

For cases $\left|i\right\rangle \neq \left|i^*\right\rangle$, the orthogonality of the states $\left|i\right\rangle$ gives $\left\langle i^*|i\right\rangle = 0$, which means that we get the expected behavior from F:

$$F\left|i\right\rangle = I\left|i\right\rangle - 0 = \left|i\right\rangle$$

For cases $\left|i\right\rangle = \left|i^*\right\rangle$, the normality of the states $\left|i\right\rangle$ means that $\left\langle i^*|i^*\right\rangle = 1$, which again gives us the expected behavior from F:

$$F\left|i^*\right\rangle = I\left|i^*\right\rangle - 2\left|i^*\right\rangle \cdot 1 = -\left|i^*\right\rangle$$

We see that the chosen representation of the operator F behaves as expected.

If F is an unitary operator, the must hold that $F^2 = I$:

$$F^2 = (I - 2\left|i^*\right\rangle\left\langle i^*\right|)^2 = I - 4\left|i^*\right\rangle\left\langle i^*\right| + 4\left|i^*\right\rangle\left\langle i^*|i^*\right\rangle\left\langle i^*\right| = I - 4\left|i^*\right\rangle\left\langle i^*\right| + 4\left|i^*\right\rangle\left\langle i^*\right| = I$$

due to the normalization of i^*, giving $\left\langle i^*|i^*\right\rangle = 1$.

For F to be hermitian, we require that $F^\dagger = F$

$$F^\dagger = (I - 2\left|i^*\right\rangle\left\langle i^*\right|)^\dagger = I^\dagger - 2(\left|i^*\right\rangle\left\langle i^*\right|)^\dagger = I - 2\left|i^*\right\rangle\left\langle i^*\right| = I$$

We see that F is both unitary and hermitian.

2.3

For $\left\langle i^*|s\right\rangle$, we get that

$$\left\langle i^*|s\right\rangle = \left\langle i^*\right|\frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left|i\right\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left\langle i^*|i\right\rangle$$

The inner product $\left\langle i^*|i\right\rangle$ will, due to the orthonormality of the $\left|i\right\rangle$s, be 0 for all cases but $\left\langle i^*|i^*\right\rangle$, where it will be 1. The sum simply evaluates to 1, giving

$$\left\langle i^*|s\right\rangle = \frac{1}{\sqrt{N}} \tag{2}$$

For $F\left|s\right\rangle$, we get that

$$F\left|s\right\rangle = (I - 2\left|i^*\right\rangle\left\langle i^*\right|)\left|s\right\rangle = I\left|s\right\rangle - 2\left|i^*\right\rangle\left\langle i^*|s\right\rangle$$

Using 2, this becomes

$$F\left|s\right\rangle = \left|s\right\rangle - \frac{2}{\sqrt{N}}\left|i^*\right\rangle$$

[2] Technically, when we have applied f $N-1$ times, we know that i^* must be the last integer, and do not need to apply f one last time. However, if we consider cases of large Ns, this case is negligible. We could also say that we *must* apply f also the N'th time, 'just to be sure' that the last value is actually i^*.

[3] We are considering the probability of needing exactly n applications of f from the start of. If we have already made $n-1$ applications of f, the probability of finding i^* at the n'th applications would be different, but this is not the probability we are looking for. Looking at the system from before any applications, each 'guess' is equally probable of giving us i^*.

2.4

We demand that $|g\rangle$ is normalized, meaning that

$$\begin{aligned}\langle g|g\rangle = 1 &= \left[\,\langle s|\,\alpha^* + \langle i^*|\,\beta^*\right] \cdot \left[\alpha\,|s\rangle + \beta\,|i^*\rangle\right] \\ &= \alpha^2\,\langle s|s\rangle + 2\alpha\beta\,\langle i^*|s\rangle + \beta^2\,\langle i^*|i^*\rangle\end{aligned}$$

Using that $|i^*\rangle$ and $|s\rangle$ are normalized, and applying the inner product from last exercise, we get that

$$\alpha^2 + \beta^2 + \frac{2}{\sqrt{N}}\alpha\beta = 1$$

which is our normalization condition.

2.5

The operator X represents a spectral representation of the eigenvalues of $|i\rangle$.

Applying the given X operator onto a state $|i\rangle$ gives us

$$X\,|i\rangle = \sum_{j=1}^{N} j\,|j\rangle\,\langle j|i\rangle$$

Due to the orthogonality of the states, only the $j = i$ term of the sum is non-zero, and we get

$$X\,|i\rangle = i\,|i\rangle$$

We see that any state $|i\rangle$ is an eigenstate of X, and gives us an eigenvalue, representing an observation of the state. Since the operator retrives the observable quantity of $|i\rangle$, we can accurately consider it an observation of i.

2.6

Since the states are non-degenerate, we can simply multiply $\langle i^*|$ from the right and square to get the probability of measuring i.

$$P(i^*) = (\langle i^*|g\rangle)^2 = (\alpha\,\langle i^*|s\rangle + \beta\,\langle i^*|i^*\rangle)^2 = \left(\frac{\alpha}{\sqrt{N}} + \beta\right)^2 = \frac{\alpha^2}{N} + 2\frac{\alpha\beta}{\sqrt{N}} + \beta^2$$

We recognize the last two terms from exercise 2.4, where we can rewrite the normalization condition as $2\frac{\alpha\beta}{\sqrt{N}} + \beta^2 = \alpha^2 - 1$, which gives

$$P(i^*) = \frac{\alpha^2}{N} + 1 - \alpha^2 = \alpha^2\left(\frac{1}{N} - 1\right) + 1$$

2.7

$$\begin{aligned}UF\,|s\rangle = U\left(|s\rangle - \frac{2}{\sqrt{N}}\,|i^*\rangle\right) &= U\,|s\rangle - \frac{2}{\sqrt{N}}U\,|i^*\rangle \\ &= 2\,|s\rangle\,\langle s|s\rangle - I\,|s\rangle - \frac{2}{\sqrt{N}}(2\,|s\rangle\,\langle s|i^*\rangle - I\,|i^*\rangle) \\ &= |s\rangle - \frac{4}{N}\,|s\rangle + |i^*\rangle = \left(1 - \frac{4}{N}\right)|s\rangle + \frac{2}{\sqrt{N}}\,|i^*\rangle\end{aligned}$$

The norm of $UF\,|s\rangle$ becomes

$$|UF\,|s\rangle\,|^2 = \langle s|\,F^\dagger U^\dagger UF\,|s\rangle = \langle s|\,FUUF\,|s\rangle$$

We can see that U is also unitary

$$U^2 = (2\,|s\rangle\,\langle s| - I)^2 = 4\,|s\rangle\,\langle s|s\rangle\,\langle s| - 4\,|s\rangle\,\langle s| + I = 4\,|s\rangle\,\langle s| - 4\,|s\rangle\,\langle s| + I = I$$

Iserting both $UU = I$ and $FF = I$, we get that

$$|UF\,|s\rangle\,|^2 = \langle s|\,FUUF\,|s\rangle = \langle s|\,FF\,|s\rangle = \langle s|s\rangle = 1$$

giving that $UF\,|s\rangle$ is normalized.

2.8

Writing out UF becomes

$$UF = (2|s\rangle\langle s| - I)(I - 2|i^*\rangle\langle i^*|) = 2|s\rangle\langle s| + 2|i^*\rangle\langle i^*| - \frac{4}{\sqrt{N}}|s\rangle\langle i^*| - I$$

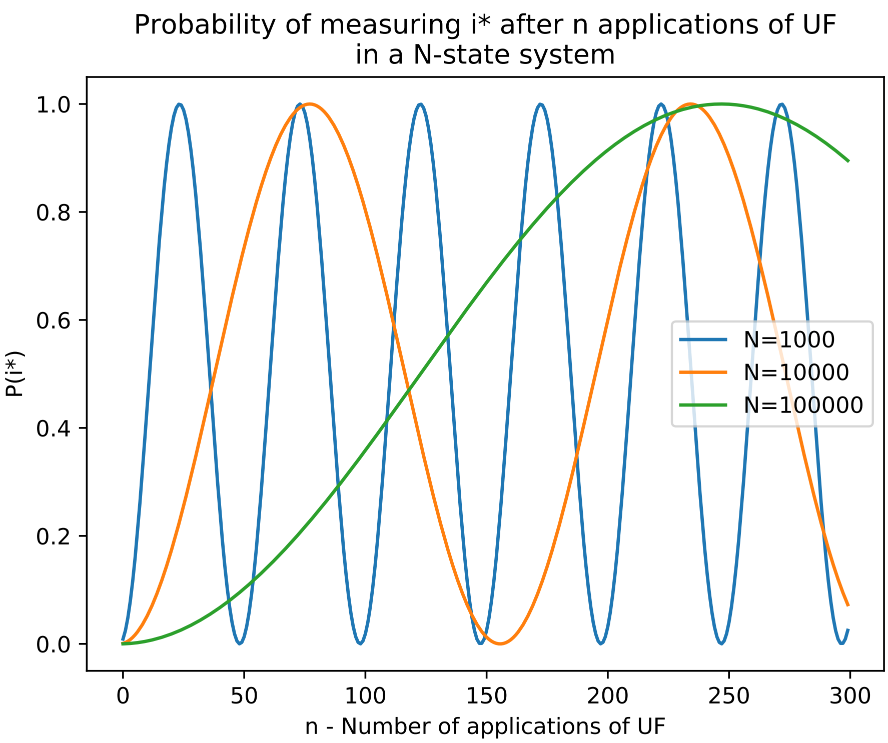
We already know that $UF|s\rangle = \left(1 - \frac{4}{N}\right)|s\rangle + |i^*\rangle$. Applying this, we get that

$$UF(\alpha|s\rangle + \beta|i^*\rangle) = \alpha UF|s\rangle + \beta UF|i^*\rangle$$
$$= \alpha\left(1 - \frac{4}{N}\right)|s\rangle + \alpha|i^*\rangle + \beta\left[2|s\rangle\langle s|i^*\rangle + 2|i^*\rangle\langle i^*|i^*\rangle - \frac{4}{\sqrt{N}}|s\rangle\langle i^*|i^*\rangle - I|i^*\rangle\right]$$
$$= \alpha\left(1 - \frac{4}{N}\right)|s\rangle + \alpha|i^*\rangle + \frac{2}{\sqrt{N}}\beta|s\rangle + 2\beta|i^*\rangle - \beta\frac{4}{\sqrt{N}}|s\rangle - \beta|i^*\rangle]$$
$$= \left[\alpha\left(1 - \frac{4}{N}\right) - \frac{2}{\sqrt{N}}\beta\right]|s\rangle + \left[\alpha\frac{2}{\sqrt{N}} + \beta\right]|i^*\rangle$$

2.9

Applying UF onto $|s\rangle$ in 2.7, we saw that the result is some linear combination of $|s\rangle$ and $|i^*\rangle$. From 2.8 we know that UF applied onto such a linear combination itself is a linear combination. If we also treat the first product, $UF|s\rangle$ as a special case of $UF|g\rangle$ with $\alpha = 1$ and $beta = 0$, this will make for a great for-loop in our program. We simply repeatidly apply UF onto the result, n times, using the formula in 2.8.

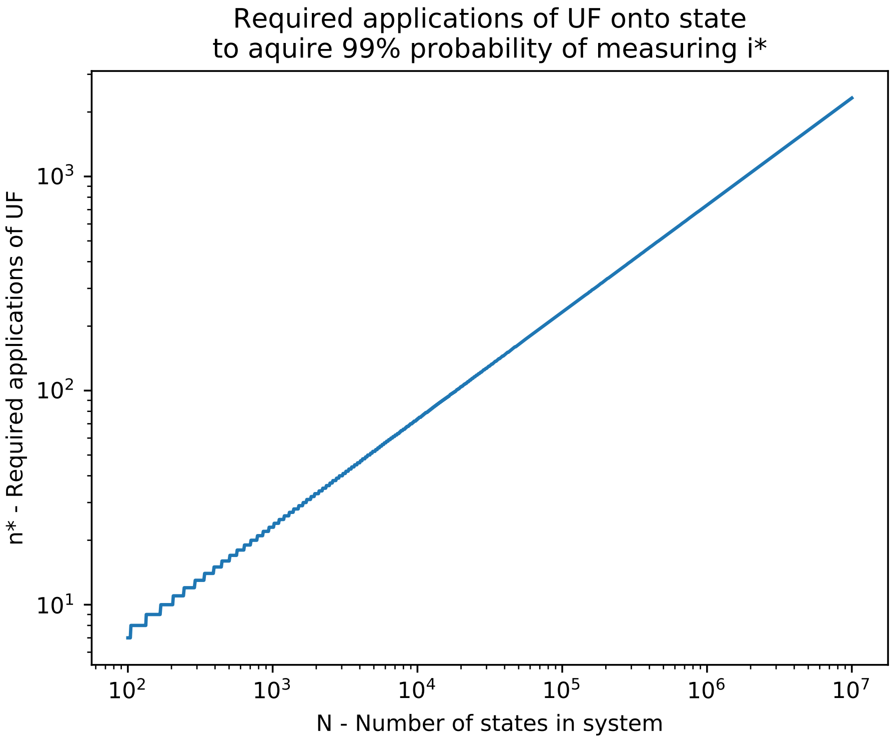
Below we see a figure showing the probability of measuring a certain state $|i^*\rangle$ after n applications of UF. As we can see, the probability oscilates between 0 and 1, with decreasing frequency as the size of the system increases.



We introduce n^* as the minumum number of n required to reach $P(i^*) \geq 0.99$ for a specific N-system. This value is shown for a set of different N's below.

N	n^*
1e2	7
1e3	23
1e4	74
1e5	233
1e6	735
1e7	2325
1e8	7353

Plotting the n^* for a large set of N's, we see a very clear pattern. n^* appears to increase linearly on a logaritmic plot, meaning that it is some exponential function of N. The inclination appears to be somewhere around 0.5, meaning $n^* \sim N^{0.5} = \sqrt{N}$.



2.10

In the last exercise, I made the guess that n^* is proportional to some exponent of N, around 0.5. Going back into our program from 2.9 and doing some linear regression of $\log(n^*)$ against $\log(N)$, we get that the slope of this linear regression is 0.500146429312. This points heavily towards the assumption that $n^* \sim \sqrt{N}$, with some small margin of error.

So what does this mean? Well, n^* is the *minimum*[4] n required for the probability of measuring a specific $|i^*\rangle$ from a set of N $|i\rangle$s becoming larger than 99%. Most systems reached far further than this (i.e. 99.99%) only a few steps after n^*. It's therefore not unreasonable to simply consider n^* the number n required for actually observing our chosen state $|i^*\rangle$.

n does itself represent the number of repeated applications of the UF operator onto the system. Since this operator is both hermitian and preserves normalization, it should be perfectly representable as a physical gate in a quantum computer. We also know from 2.1 that we will be capable of differentiating the $|i^*\rangle$ case from the rest of the states through the F operator.

This means that for a set of N $|i\rangle$s (which can represent a database of some values), we can find *any* specific state $|i^*\rangle$ by passing our system through some gate a number of times that scales with $\sqrt{N}$. It's natural to assume that "passing our system through the gate" takes some constant amount of time. We then have a computer which can look for a value i^* in a database of size N in a time that scales with $\sqrt{N}$.

[4]In our calculations of n^*, we have chosen our i^* to be the least likely observed state at the begining, and thus the slowest state to observe. We therefore know that no chosen i^* would be slower to observe than n^*.

Appendix - 2.9 Code

```
import matplotlib.pyplot as plt
from numpy import zeros, logspace, int64, log10
from math import sqrt
from scipy import stats

def UF(alpha, beta):
    # Calculates UF(alpha|s> + beta|i*>) = s_coeff|s> + i_coeff|i*>
    # and returns the new coefficients in front of s and i*.
    s_coeff = alpha*(1- 4/N) - 2/sqrt(N)*beta
    i_coeff = alpha*2/sqrt(N) + beta
    return s_coeff, i_coeff

def condition(alpha, beta):
    # Returns the normalization condition on alpha and beta, which should be 1.
    return alpha**2 + beta**2 + 2*alpha*beta/sqrt(N)

def prob_alpha(alpha, N):
    # Returns the probability of measuring a state i*
    return alpha**2*(1/N - 1) + 1

# Plotting the probability for the three Ns.
nr_of_ns = 300
for N in [int(1e3), int(1e4), int(1e5)]:
    alpha = 1
    beta = 0
    prob_array = zeros(nr_of_ns)
    for n in range(nr_of_ns):
        alpha, beta = UF(alpha, beta)
        prob_array[n] = prob_alpha(alpha, N)
    plt.plot(prob_array, label="N=%d" % N)
plt.title("Probability of measuring i* after n applications of UF\nin a N-state system")
plt.xlabel("n - Number of applications of UF")
plt.ylabel("P(i*)")
plt.legend()
plt.savefig("three_Ns.pdf")
plt.clf()

print("%10s%10s" % ("N", "n*"))
for N in [int(1e2), int(1e3), int(1e4), int(1e5), int(1e6), int(1e7), int(1e8)]:
    alpha = 1
    beta = 0
    n_star = 0
    prob = 0
    while prob < 0.99:
        alpha, beta = UF(alpha, beta)
        prob = prob_alpha(alpha, N)
        n_star += 1
    print("%10.0e%10s" % (N, n_star))

Ns = logspace(2, 7, 1000, dtype=int64)
n_stars = zeros(len(Ns))
for i in range(len(Ns)):
    N = Ns[i]
    alpha = 1
    beta = 0
    n_star = 0
    prob = 0
    while prob < 0.99:
        alpha, beta = UF(alpha, beta)
```

```
        prob = prob_alpha(alpha, N)
        n_star += 1
    n_stars[i] = n_star

slope, intercept, r_value, p_value, std_err = stats.linregress(log10(Ns), log10(n_stars))

print("Slope of the logaritmic curve = ", slope)

plt.loglog(Ns, n_stars)
plt.title("Required applications of UF onto state\nto aquire 99% probability of measuring i*")
plt.xlabel("N - Number of states in system")
plt.ylabel("n* - Required applications of UF")
plt.savefig("n_star.pdf")
plt.clf()
```