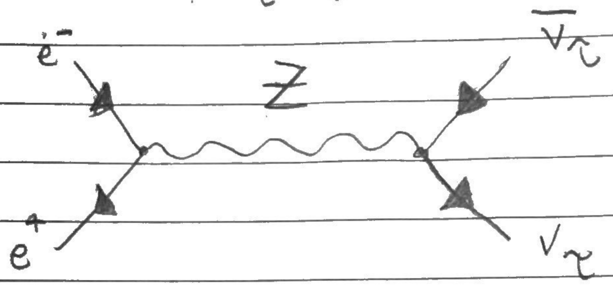
1)

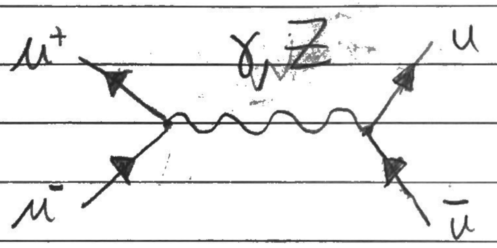
PSET 12 Electroweak Bosons W, Z, and γ

1a) $e^+e^- \to \nu_\tau \bar{\nu}_\tau$

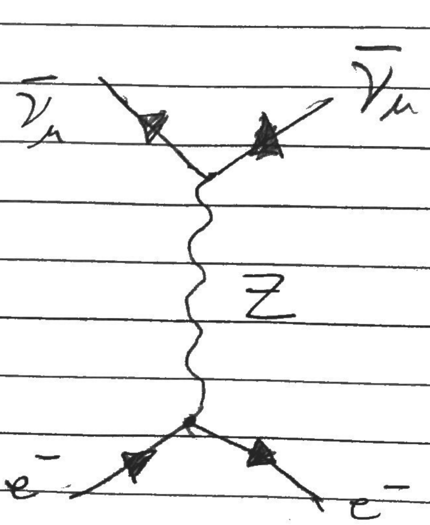


γ doesn't work b/c products aren't charged

1b) $\mu^+\mu^- \to u\bar{u}$, u = up quark.



1c) $\bar{\nu}_\mu + e^- \to \bar{\nu}_\mu + e^-$

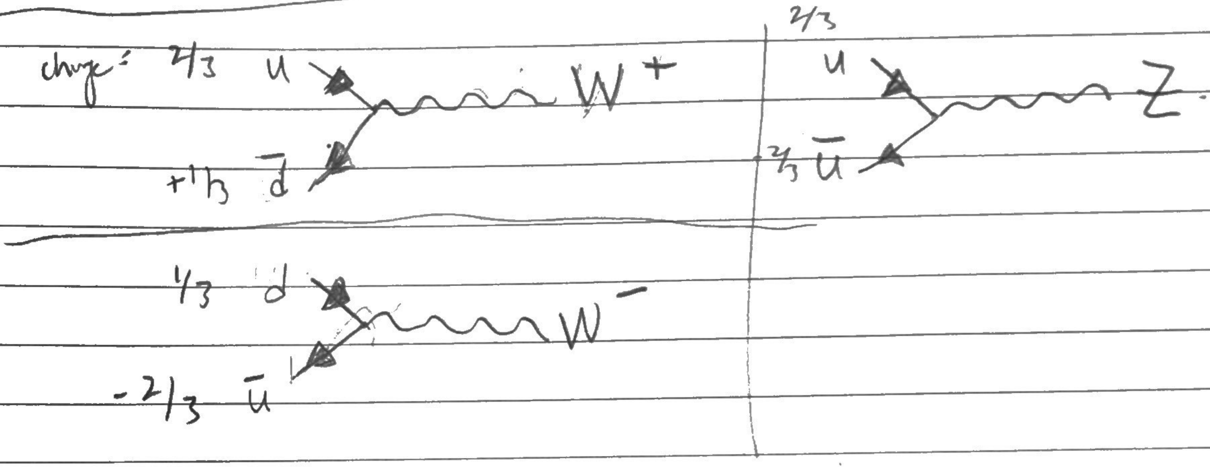


$\cancel{1eV} \cdot \frac{1GeV}{\cancel{1eV}}$

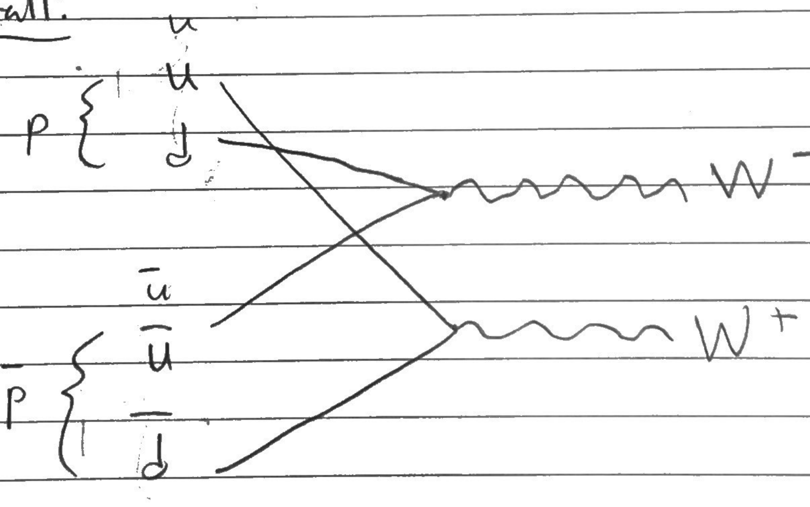
wrong order

1e) $p\bar{p} \to W^+W^-$

$\bar{p}$: $\bar{u}\bar{u}\bar{d}$, p: uud

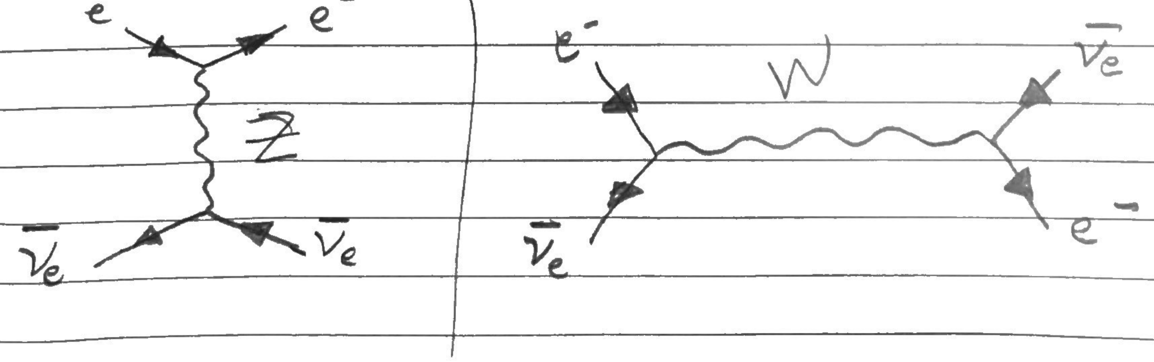


Overall:



1d — wrong order

$\bar{\nu}_e + e^- \to \bar{\nu}_e + e^-$



Problem Set 12 problems

Question 1: Feynman diagrams with γ, W and Z bosons (30 points)

Learning objectives

In this question you will:

- Apply knowlege of Feynman diagrams involved electroweak bosons to various physical processes

Draw the lowest order Feynman diagram or diagrams for each of the following processes

1a.

$e^+e^- \to \nu_\tau\overline{\nu}_\tau$

Write your answer here

1b.

$\mu^+\mu^- \to u\overline{u}$ where u is an up quark

Write your answer here

1c.

$\overline{\nu}_\mu + e^- \to \overline{\nu}_\mu + e^-$

Write your answer here

1d.

$\overline{\nu}_e + e^- \rightarrow \overline{\nu}_e + e^-$

Write your answer here

1e.

$p\overline{p} \rightarrow W^+W^-$ (consider only the valence quarks in the proton and anti-proton)

Write your answer here

Question 2: Mixing in the K and B systems (30 points)

Learning objectives

In this question you will:

- Review the formulae for flavor mixing and see how the mixing phenomenology depends on $\Delta\Gamma$ and Δm

The phenomenon if K^0-$\overline{K}^0$ mixing was discussed in Lecture 22 and in Thomson Section 14.5. If a K^0 is produced via the strong interactions at $t = 0$ the probabilities of the state being observed as a K^0 or as a $\overline{K}^0$ at a later time t are giving by the expressions

$$\begin{aligned} P(K^0_{t=0} \rightarrow K^0) &= \tfrac{1}{4}\left[e^{-\Gamma_1 t} + e^{-\Gamma_2 t} + 2e^{-(\Gamma_1+\Gamma_2)t/2}\cos(\Delta m t)\right] \\ P(K^0_{t=0} \rightarrow \overline{K}^0) &= \tfrac{1}{4}\left[e^{-\Gamma_1 t} + e^{-\Gamma_2 t} - 2e^{-(\Gamma_1+\Gamma_2)t/2}\cos(\Delta m t)\right] \end{aligned}$$

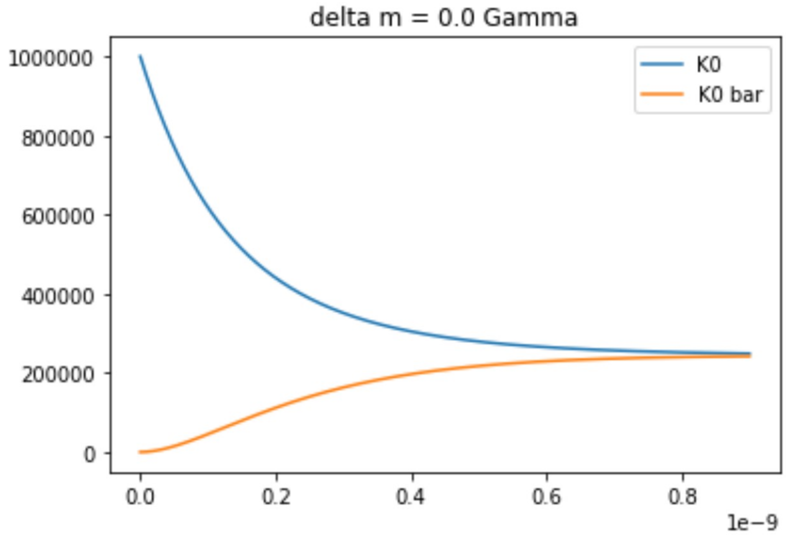
where Γ_1 and Γ_2 are the inverse lifetimes of the K_S and K_L states and Δm is the mass difference between the states. Note: for this problem we will ignore CP violation so that $|K_S> \equiv |K_1>$ and $|K_L> \equiv |K_2>$

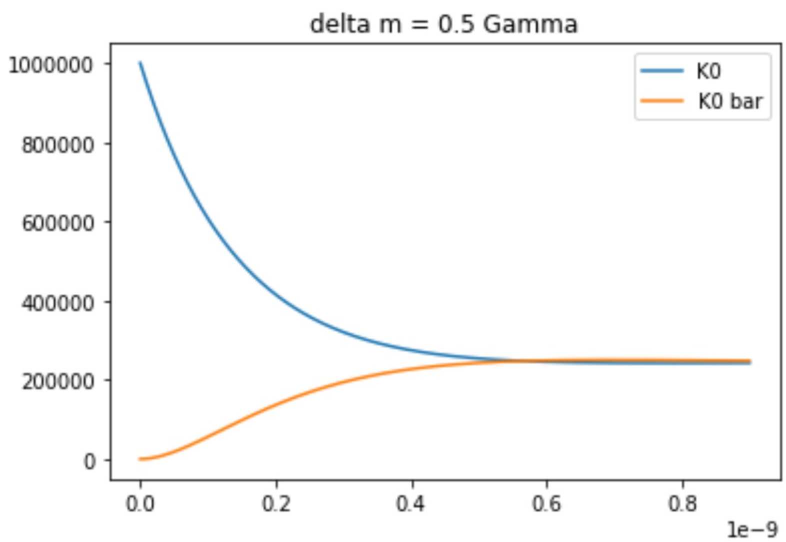
2a.

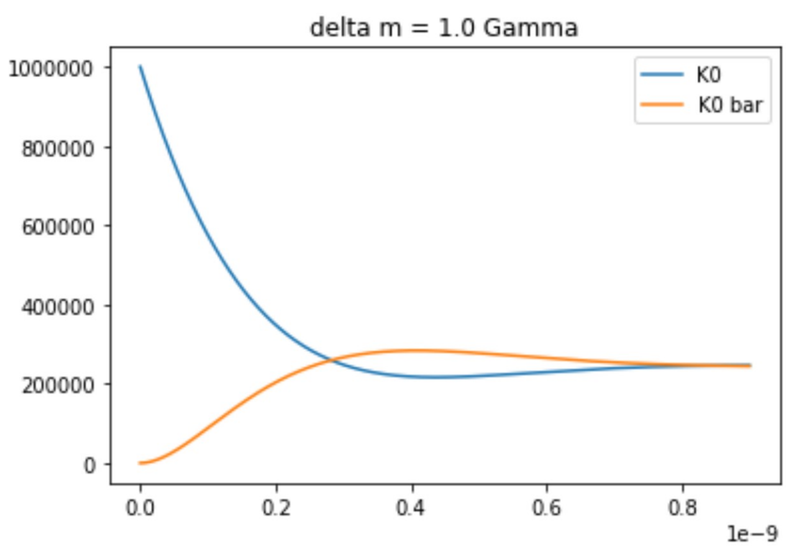
If a beam of 1 million K^0 is produced, using the measured values of the K_S and K_L lifetimes plot the predicted number of K^0 and of $\overline{K}^0$ that would be present in the beam as a function of time for $0 < t < \frac{10}{\Gamma_1}$ if the value of Δm were 0, 0.5Γ or Γ.

In [8]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [9]:
```
num_K0 = 10**6

#from slides:
tau_s = 0.9*10**(-10) #sec
tau_p = 0.5*10**(-7) #sec

gamma1 = 1/tau_s
gamma2 = 1/tau_p

gamma = (gamma1 + gamma2)/2
deltam = np.array([0, 0.5, 1])*gamma


times = np.linspace(0, 10/gamma1, 1000)

def KtoK(t, g1, g2, dm, bar = False):
    a = np.exp(-1*g1*t)
    b = np.exp(-1*g2*t)
    c = 2*np.exp(-1*(g1 + g2)*t/2)*np.cos(dm*t)

    if bar == True:
        c *= -1

    return (1/4)*(a + b + c)


for i in range(len(deltam)):
    K0s = KtoK(times, gamma1, gamma2, deltam[i])*num_K0
    K0bars = KtoK(times, gamma1, gamma2, deltam[i], bar = True)*num_K0

    titlestr = str(i*0.5) + " Gamma"

    plt.figure()
    plt.title("delta m = " + titlestr)
    plt.plot(times, K0s, label = "K0")
    plt.plot(times, K0bars, label = "K0 bar")
    plt.legend()

plt.show()
```


delta m = 0.0 Gamma
K0
K0 bar
1000000
800000
600000
400000
200000
0
0.0
0.2
0.4
0.6
0.8
1e−9

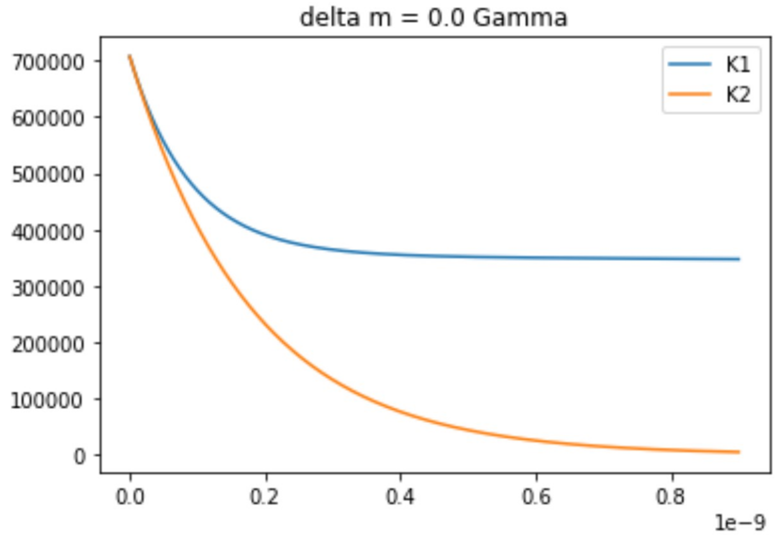

delta m = 0.5 Gamma
K0
K0 bar
1000000
800000
600000
400000
200000
0
0.0
0.2
0.4
0.6
0.8
1e−9

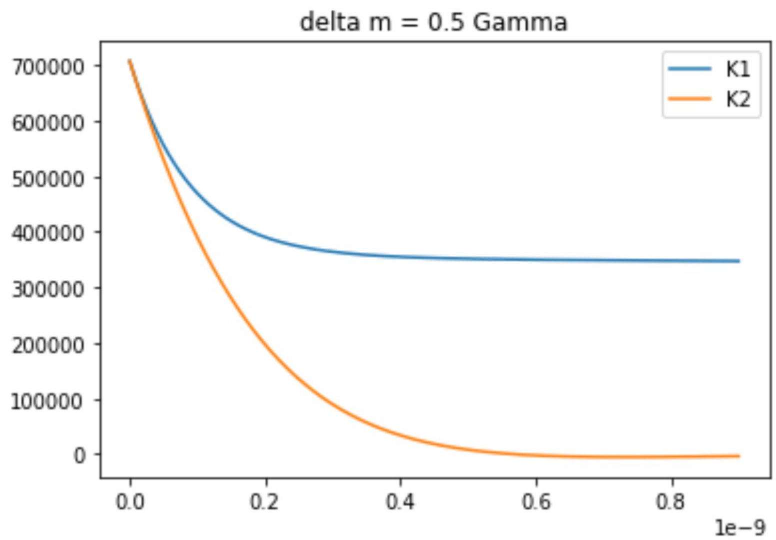

delta m = 1.0 Gamma
K0
K0 bar
1000000
800000
600000
400000
200000
0
0.0
0.2
0.4
0.6
0.8
1e−9

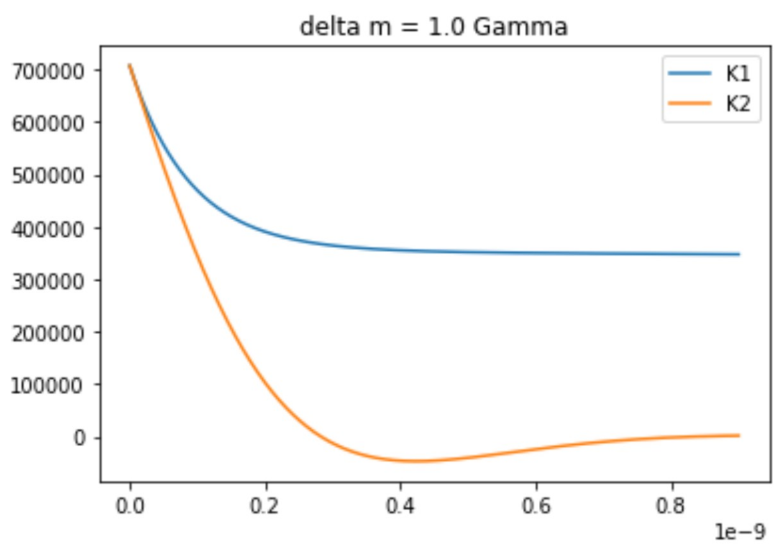
2b.

Remake the plot above but plotting the predicted number of K_1 and K_2 in the beam instead of K_0 and $\overline{K}_0$

In [10]:
```
# by definition:

for i in range(len(deltam)):
    K0s = KtoK(times, gamma1, gamma2, deltam[i])*num_K0
    K0bars = KtoK(times, gamma1, gamma2, deltam[i], bar = True)*num_K0

    K1s = (1/np.sqrt(2))*(K0s + K0bars)
    K2s = (1/np.sqrt(2))*(K0s - K0bars)

    titlestr = str(i*0.5) + " Gamma"

    plt.figure()
    plt.title("delta m = " + titlestr)
    plt.plot(times, K1s, label = "K1")
    plt.plot(times, K2s, label = "K2")
    plt.legend()

plt.show()
```


delta m = 0.0 Gamma
K1
K2
700000
600000
500000
400000
300000
200000
100000
0
0.0
0.2
0.4
0.6
0.8
1e−9


delta m = 0.5 Gamma
K1
K2
700000
600000
500000
400000
300000
200000
100000
0
0.0
0.2
0.4
0.6
0.8
1e−9


delta m = 1.0 Gamma
K1
K2
700000
600000
500000
400000
300000
200000
100000
0
0.0
0.2
0.4
0.6
0.8
1e−9

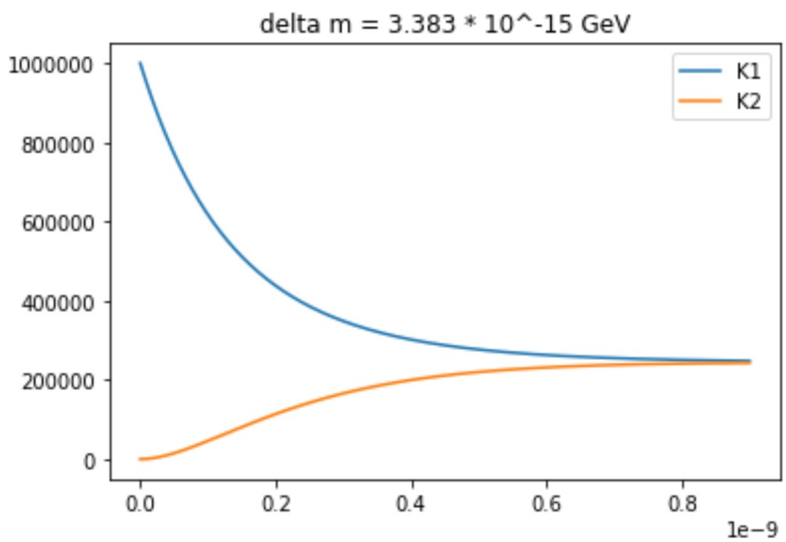
2c.

The measured value of the mass difference is

$$\Delta m = 3.383 \times 10^{-15} \text{ GeV}$$

remake the plots above for this value

In [20]:
```python
given_dm = 3.383*(10**(-15)) # Gev

### convert to freq
h = 4.135*(10**(-15)) #eV*s
hGeV = h*(10**(-9)) # GeV*s

converted_dm = given_dm/hGeV #s

K1s = KtoK(times, gamma1, gamma2, converted_dm)*num_K0
K2s = KtoK(times, gamma1, gamma2, converted_dm, bar = True)*num_K0

titlestr = "3.383 * 10^-15 GeV"

plt.figure()
plt.title("delta m = " + titlestr)
plt.plot(times, K1s, label = "K1")
plt.plot(times, K2s, label = "K2")
plt.legend()

plt.show()
```

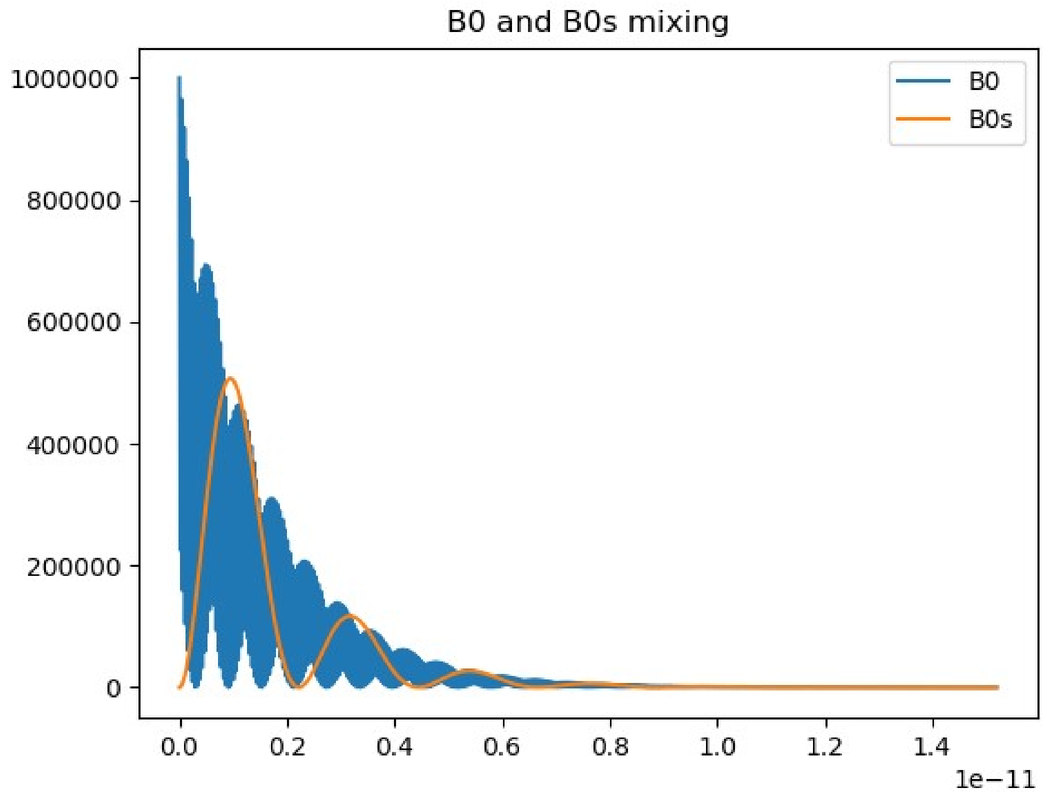


2d

The same expressions used for K^0 mixing can be applied to the B^0 and to the B_s. Use the PDG to find the values of Γ_1, Γ_2 and Δm for the B^0 and the B_s. Using these numbers repeat 2c for these two neutral mesons.

In [26]:
```python
%matplotlib notebook
```

In [28]:
```
# convention: 1 = B0, 2 = B0s
m_B0 = 5279.61 # MeV
m_B0s = 5366.79 # MeV

tau_B0 = 1.520*10**(-12) #s
tau_B0s = 1.510*10**(-12) #s

gammaB0 = 1/tau_B0
gammaB0s = 1/tau_B0s

deltam_B = (m_B0 - m_B0s)/2

deltam_B0s = 1.1688*10**(-8) #MeV
deltam_B0 = 0.32 #MeV

hMeV = h*(10**(-6)) #MeV*s
deltam_B0s /= hMeV #s
deltam_B0 /= hMeV #s

new_times = np.linspace(0, 10/gammaB0, 1000)

B0 = KtoK(new_times, gammaB0, gammaB0s, deltam_B0)*num_K0
B0s = KtoK(new_times, gammaB0, gammaB0s, deltam_B0s, bar = True)*num_K0


plt.figure()
plt.title("B0 and B0s mixing")
plt.plot(new_times, B0, label = 'B0')
plt.plot(new_times, B0s, label = 'B0s')
plt.legend()


plt.show()
```


B0 and B0s mixing
B0
B0s
1000000
800000
600000
400000
200000
0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1e−11

In [ ]: