Vedaank Tiwari
11-13-17
MW5-6

EE16A Homework HW 10

1

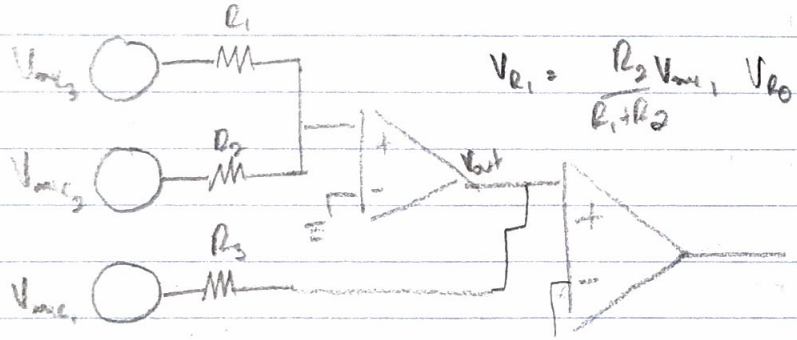
A $\vec{S}_{sound} = A \vec{S}_{mic}$

$(2\times 3)(2\times 1)$

A = 2x3 matrix | $A = \begin{bmatrix} a_1 left & a_2 left & a_3 left \\ a_1 right & a_2 right & a_3 right \end{bmatrix}$

B $S_{ear} = S_{sound} = \begin{bmatrix} a_1 left & a_2 left & a_3 left \\ a_1 right & a_2 right & a_3 right \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} V_{mic1} \\ V_{mic2} \\ V_{mic3} \end{bmatrix}$

D



$V_{R_1} = \frac{R_2 V_{mic1}}{R_1 + R_2}$, $V_{R_2} = \frac{R_1 V_{mic2}}{R_1 R_2}$

$V_{R_3} = \frac{V_{out} - V_{mic}}{I_R}$

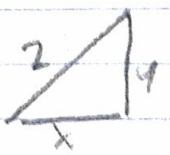
C $\vec{S}_{source} = \begin{bmatrix} s_{left} \\ s_{right} \end{bmatrix}$

$S_{cancel} \cdot S_{rec} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ | $S_{cancel} = [1 \; 0]$

2 $|\langle \vec{x}, \vec{y} \rangle| = |\vec{x}^T \vec{y}| \leq \|\vec{x}\| \cdot \|\vec{y}\|$

$\|\vec{x} + \vec{y}\| \leq \|\vec{x}\| + \|\vec{y}\|$

$\|x + y\|^2$.



Details of triangle = $x + y > z$

$z(\vec{x} + \vec{y}), (\vec{x} + \vec{y}) > = \vec{x}(\vec{x} + \vec{y}) + \vec{y}(\vec{x} + \vec{y})$

$= \|\vec{x}\|^2 + 2\vec{x}\vec{y} + \|\vec{y}\|^2$

3

A $x_1 = [1\ 1\ 1 \dots 1]^T \quad x_2 = [1\ -1\ 1 \dots -1]^T$

$(x_1)(x_2) = \boxed{\frac{N}{2}}$

B $\vec{x} = [-1\ 1\ 1\ -1\ 1\ 1\ -1\ 1\ \ ]$

$\vec{x} = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ x_8]$

$\vec{y} = [1\ 1\ -1]^T \quad y = [1\ 1\ 1^T]$

$[x_i\ x_{i+1}\ x_{i+2}]^T$ Best I = 2, 5

We can use a correlation matrix.

C $\vec{y} = [1\ 2\ 3]^t \quad \vec{x} = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]$

$\vec{X} = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \\ 4 & 5 & 6 \\ 5 & 6 & 7 \\ 6 & 7 & 8 \\ 7 & 8 & 1 \\ 8 & 1 & 2 \end{bmatrix}$

$X \cdot y^T =$ see if any equal $\vec{y} \cdot \vec{y}$

← use this correlation matrix

D ipython

E ipython

4

A Yes this is valid as generally the ground placement doesn't make a significant difference. It is merely a reference point

B Yes, v_4 and v_3 are identical, as they are connected by an ideal wire, so they have the same current, voltage, etc.

C No, v_4 is labeled incorrectly. Voltage drop from + to −, which is not passive sign convention

5

A $V_1 = \frac{V_1}{V_1 + V_2} = \frac{4}{4+4} = \frac{1}{8} \cdot 12V = 6V$

B $R_{Th} = \frac{4}{4+4} = \frac{1}{2}$

$V_{Th} = 12V$

C $R_{No} = \frac{2V}{\frac{1}{\frac{1}{4}+\frac{1}{4}} + 2} = \frac{2}{8+2} = \boxed{\frac{1}{5}}$

$I_{No} = \frac{12}{1/5} = \boxed{60 \text{ Amps}}$ | $V = IR$ $I = \frac{V}{R}$

6

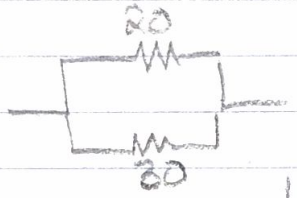
A $\rho = R\frac{A}{l}$ $\quad R = \frac{\rho l}{A}$

$1\times10^{-8}\,\Omega \cdot \frac{75\times10^{-9}}{(5\times10^{-9})^2} = \frac{75\times10^{-9}}{25\times10^{-18}} = 3\times10^{9}\times10^{-8} =$

$\boxed{30}$

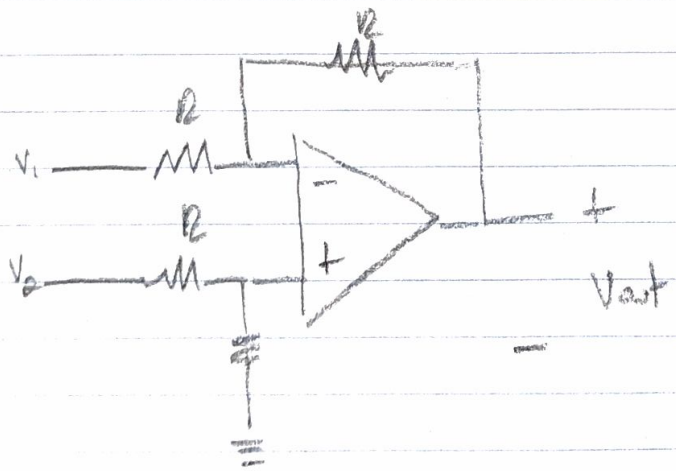
B $2\times10^{-8}\,\Omega \cdot \frac{75\times10^{-9}}{(10\times10^{-9})^2 - (5\times10^{-9})^2} = 2\times10^{-8}\,\frac{75\times10^{-9}}{75\times10^{-18}} = \boxed{20}$

C



D $R_{eq} = \frac{1}{\frac{1}{20}+\frac{1}{30}} = \frac{1}{\frac{30}{600}+\frac{20}{600}} = \frac{600}{50} = \boxed{12}$

7



$V_+ = V_-$

$V_{out} = \boxed{\frac{1}{2}V_1}$ $\quad V_2$ does not matter

6

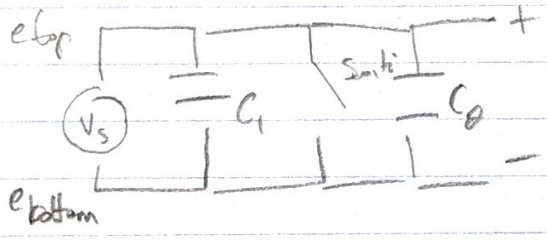
A $V_{press} = \frac{R_{touch}}{R_{mid} + R_{touch}} \cdot V_S$ Incorrect

B $V_{press} = \frac{R_{bottom} + R_{meter}}{R_{top} + R_{mid} + R_{mid}} V_S$

9

A $C_{notouch} = \frac{\varepsilon A}{d} = \boxed{\frac{\varepsilon_1 A}{d}}$

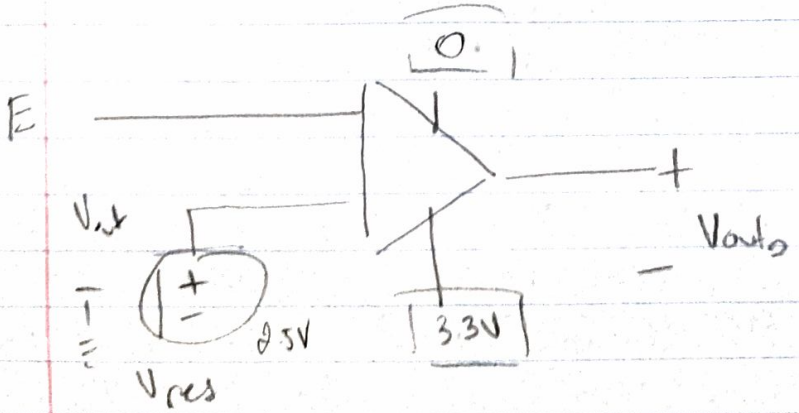
B



C $C_{press} = \boxed{\frac{\varepsilon_1 A}{d'}}$

D $V_{out} = \frac{C_{screen}}{C_{screen} + C_{ref}} V_S$

E



10

A $I_{RL} = \frac{V_s}{R_{eq}}$

$V_- = V_+ \quad I^- = 0$

$R_{eq} = \frac{R_L}{R_s + R_L}$

$V_- = V_s \cdot \frac{R_L}{R_L + R_s}$

B $I_{RL} = \frac{V_s}{R_s}$

C $I_{RL} = \frac{V_s}{R_s}$

D $V_- = \frac{V_s R_L}{R_L + R_s}$

b = $V_{out} = \frac{-V_s R_L}{R_L + R_s}$

10 A $R_{speal} = -1000$

B. $V_{out} = \frac{500}{500 + R_{spec}} = \frac{500}{-500} = -V_s$

10C , yes negative feedback
ground → +
Vout → V−

$V_{out} = \frac{R_{spec}}{500 + R_{spec}} =$

10D $\frac{-R_{spec}}{500 + R_{spec}} = \frac{-1000}{-500} = 2$

Incorrect

(11) How to check if vectors are orthogonal.

$\vec{U} \cdot \vec{V} = 0$

(12) Worke alone

EE16A Homework 10

Question 3: Audio File Matching

This notebook continues the audio file matching problem. Be sure to have `song.wav` and `clip.wav` in the same directory as the notebook.

In this notebook, we will look at the problem of searching for a small audio clip inside a song.

If you have trouble playing the audio file in IPython, try opening it in a different browser. There were problems with Safari, but Chrome works fine.

In [9]:
```
import numpy as np
import wave
import matplotlib.pyplot as plt
import scipy.io.wavfile
import operator
from IPython.display import Audio
%matplotlib inline

given_file = 'song.wav'
target_file = 'clip.wav'
rate_given,  given_signal  = scipy.io.wavfile.read(given_file)
rate_target, target_signal = scipy.io.wavfile.read(target_file)
given_signal  = given_signal[:2000000].astype(float)
target_signal = target_signal.astype(float)
def play_clip(start, end, signal=given_signal):
    scipy.io.wavfile.write('temp.wav', rate_given, signal[start:end].astype(np.i
    return Audio(url='temp.wav', autoplay=True)

def run_comparison(target_signal, given_signal, idxs=None):
    # Run everything if not called with idxs set to something
    if idxs is None:
        idxs = [i for i in range(len(given_signal)-len(target_signal))]
    return idxs, [vector_compare(target_signal, given_signal[i:i+len(target_sigr
                for i in idxs]

play_clip(0, len(given_signal))

# scipy.io.wavfile.write(target_file, rate_given, (-0.125*given_signal[1380000:1
```

Out[9]:

We will load the song onto the variable `given_signal` and load the short clip onto the variable `target_signal`. Your job is to finish the code that will identify the short clip's location in the song.

The clip we are trying to find will play after executing the following block.

In [10]:
```
Audio(url=target_file, autoplay=True)
```

Out[10]:

Your task is to define the function `vector_compare` and run the following code. Because the song has a lot of data, you should use the provided examples from the previous parts of the problem before running the later code. Do your results here make sense given your answers to previous parts of the problem?

Part (a)

In [20]:
```
def vector_compare(desired_vec, test_vec):
    """This function compares two vectors, returning a number.
    The test vector with the highest return value is regarded as being closest t
    # Hint: Use transpose for the first argument of np.dot
    # YOUR CODE HERE
    return np.dot(np.transpose(desired_vec), test_vec)

print(vector_compare(np.array([1,1,1]), np.array([1,1,1])))
print(vector_compare(np.array([1,1,1]), np.array([-1,-1,-1])))
```

```
3
-3
```

Part (c)

In [21]:
```
print(vector_compare(np.array([1,2,3]), np.array([1,2,3])))
print(vector_compare(np.array([1,2,3]), np.array([2,3,4])))
print(vector_compare(np.array([1,2,3]), np.array([3,4,5])))
print(vector_compare(np.array([1,2,3]), np.array([4,5,6])))
print(vector_compare(np.array([1,2,3]), np.array([5,6,7])))
print(vector_compare(np.array([1,2,3]), np.array([6,7,8])))
```

```
14
20
26
32
38
44
```

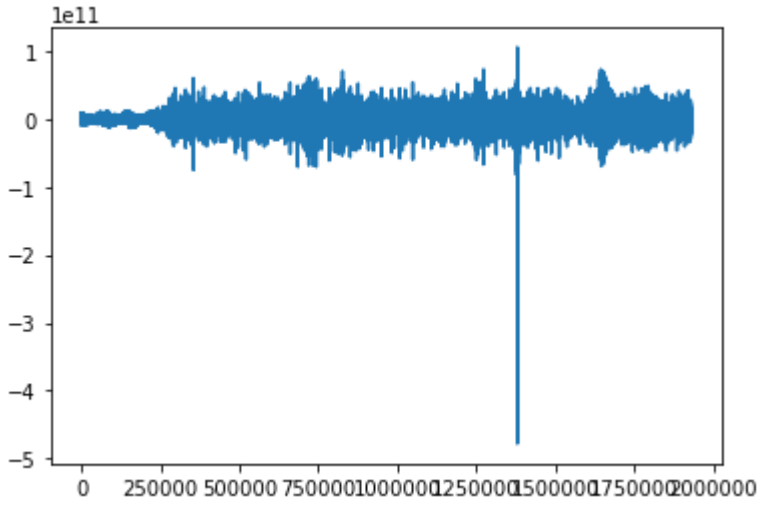
Part (d)

Run the following code that runs `vector_compare` on every subsequence in the song - it will probably take at least 5 minutes. How do you interpret this plot to find where the clip is in the song?

In [22]:
```
import time

t0 = time.time()
idxs, song_compare = run_comparison(target_signal, given_signal)
t1 = time.time()
plt.plot(idxs, song_compare)
print ("That took %(time).2f minutes to run" % {'time':(t1-t0)/60.0} )
```

```
That took 0.79 minutes to run
```



Part (e)

In the space below, write code that uses `song_compare` to print the index of `given_signal` where `target_signal` begins. Then, verify that your answer is correct by playing the song at that index using the `play_clip` function.

In [24]:
```
np.argmax(song_compare)
```

Out[24]:
```
1380284
```

In [ ]:
```

```