Final Exam for EE2703 (20 Marks) **May 20, 2021**

Dept. of Electrical Engineering, IIT Madras
Applied Programming Lab Jan 2021 session

- ▷ **vector operations are a must or lose lots of marks!!**
- ▷ **Label all plots. Add legends. Make the plots professional looking.**
- ▷ **Comments are not optional. They are required.**
- ▷ **pseudocode should be readable and neatly formatted.**
- ▷ **PDF file should be named *your-roll-number.pdf***
- ▷ **Python code should be named *your-roll-number.py* Please note that I will accept only raw python code and it should run in Python 3.x (prefer Python 3.8) So don't send me Jupyter notebooks.**
- ▷ **Python code should run!!**
- ▷ **Pdf file should include all plots and tables.**
- ▷ **The Pdf should be submitted to the 'final' assignment, and the .py code should be submitted to the 'final-code' assignment.**

This is a problem in radiation from a loop antenna of length λ.

A long wire carries a current

$$I = \frac{4\pi}{\mu_0}\cos(\phi)\exp(j\omega t)$$

through a loop of wire. Here, ϕ is the angle in polar coordinates i.e., in (r,ϕ,z) coordinates. The wire is on the $x-y$ plane and centered at the origin. The radius of the loop is 10 cm and is also equal to $1/k = c/\omega$. (This means that the circumference is λ)

The problem is to compute and plot the magnetic field $\vec{B}$ along the z axis from 1cm to 1000 cm, plot it and then fit the data to $\left|\vec{B}\right| = cz^b$. The main challenge in this exam is to handle Python arrays efficiently, and also to understand what accuracy a particular choice of grid will give.

The computation involves the calculation of the vector potential

$$\vec{A}(r,\phi,z) = \frac{\mu_0}{4\pi}\int \frac{I(\phi)\hat{\phi}e^{-jkR}ad\phi}{R}$$

where $\vec{R} = \vec{r}-\vec{r}'$ and $k = \omega/c = 0.1$. $\vec{r}$ is the point where we want the field, and $\vec{r}' = a\hat{r}'$ is the point on the loop. Due to the This can be reduced to a sum:

$$\vec{A}_{ijk} = \sum_{l=0}^{N-1} \frac{\cos(\phi_l')\exp(-jkR_{ijkl})\,\vec{dl}'}{R_{ijkl}} \tag{1}$$

where $\vec{r}$ is at r_i, ϕ_j, z_k and $\vec{r}'$ is at $a\cos\phi_l'\hat{x} + a\sin\phi_l'\hat{y}$. Note that Eq. (1) is valid for any $\left(x_i, y_j, z_k\right)$, and is summed over the current elements in the loop. You must implement this as a vector operation over both l and over a vector of $\left(x_i, y_j, z_k\right)$ values.

From $\vec{A}$, you can obtain $\vec{B}$ as

$$\vec{B} = \nabla\times\vec{A}$$

Along the z axis this becomes ($\vec{A}$is along $\hat{\phi}$ and the curl gives only aB_z component along $\hat{z}$.

$$B_z(z) = \frac{A_y(\Delta x,0,z) - A_x(0,\Delta y,z) - A_y(-\Delta x,0,z) + A_x(0,-\Delta y,z)}{4\Delta x\Delta y} \tag{2}$$

1. **[4 marks] Write pseudocode for how you will solve this problem.**

2. **[2 marks] Break the volume into a 3 by 3 by 1000 mesh, with mesh points separated by 1cm. The 3 by 3 grid in $x-y$ is needed to compute the curl using Eq. 2.**

3. **[6 marks] Break the loop into 100 sections. Plot the current elements in $x-y$ (place points at the centre points of the elements. Properly label the graph.**

4. **[4 marks] (may be done earlier)Obtain the vectors $\vec{r}'_l$, $\vec{dl}_l$, where l indexes the segments of the loop.**

5. **[6 marks] Define a function `calc(l)` that calculates and returns $\vec{R}_{ijkl} = \left|\vec{r}_{ijk} - \vec{r}'_l\right|$ for all $\vec{r}_{ijk}$ (l is the index into the $\vec{r}'$ array, which you have defined earlier.) Note: vectorize this function!**

6. **[4 marks] Extend `calc` to generate the terms in Eq. 1 in the sum and return the term to add to $\vec{A}$. Note: vectorize the function**

7. **[4 marks] Use the function to compute $\vec{A}_{ijk}$ (you can use a for loop here. Justify in a comment)**

8. **[4 marks] Now compute $\vec{B}$ along the z axis. Use Eq. 2; remember to vectorize.**

9. **[4 marks] Plot the magnetic field $B_z(z)$. Use a log-log plot.**

10. **[6 marks] Fit the field to a fit of the type $B_z \approx cz^b$.**

11. **[6 marks] Discuss your finding. Does B_z fall off as expected? What decay rate would you have expected for a static magnetic field? Where is the difference coming from?**

Useful Python Commands (use “?” to get help on these from ipython)

```
from pylab import *
import system-function as name
Note: lstsq is found as scipy.linalg.lstsq
ones(List)
zeros(List)
range(N0,N1,Nstep)
arange(N0,N1,Nstep)
linspace(a,b,N)
logspace(log10(a),log10(b),N)
X,Y=meshgrid(x,y)
where(condition)
where(condition & condition)
where(condition | condition)
a=b.copy()
lstsq(A,b) to fit A*x=b
A.max() to find max value of numpy array (similalry min)
A.astype(type) to convert a numpy array to another type (eg int)
def func(args):
  ...
  return List
matrix=c_[vector,vector,...] to create a matrix from vectors
figure(n) to switch to, or start a new figure labelled n
plot(x,y,style,...,lw=...)
semilogx(x,y,style,...,lw=...)
semilogy(x,y,style,...,lw=...)
loglog(x,y,style,...,lw=...)
contour(x,y,matrix,levels...)
quiver(X,Y,U,V) # X,Y,U,V all matrices
xlabel(label,size=)
ylabel(label,size=)
title(label,size=)
xticks(size=) # to change size of xaxis numbers
yticks(size=)
legend(List) to create a list of strings in plot
annotate(str,pos,lblpos,...) to create annotation in plot
```