FFT

Calculate the FFT of a time domain signal
Total: 12 marks

From the time signal that you are given, you must calculate the FFT. Indicate the steps that you followed. (3 marks) *

- [ ] Normalize the FFT by multiplying it by 2/N
- [ ] Do not normalize the FFT
- [ ] Do not apply windowing
- [ ] Do not perform FFT correction
- [ ] Apply windowing before calculating the FFT
- [ ] Perform FFT correction by multiplying the FFT with the window correction factor
- [ ] Apply windowing after calculating the FFT
- [ ] Perform FFT correction by dividing the FFT with the window correction factor
- [ ] Other: ____________

Enter the frequency of the 1st positive frequency peak (1 mark) *

Your answer

Enter the frequency of the 2nd positive frequency peak (1 mark) *

Your answer

Enter the frequency of the 3rd positive frequency peak (1 mark) *

Your answer

Enter the amplitude of the 1st positive frequency peak (**accurate to 3 significant figures**) (1 mark) *

Your answer

Enter the amplitude of the 2nd positive frequency peak (**accurate to 3 significant figures**) (1 mark) *

Your answer

Enter the amplitude of the 3rd positive frequency peak (**accurate to 3 significant figures**) (1 mark) *

Your answer

Enter the phase angle (**in degrees**) of the 1st positive frequency peak (1 mark) *

Your answer

Enter the phase angle (**in degrees**) of the 2nd positive frequency peak (1 mark) *

Your answer

Enter the phase angle (**in degrees**) of the 3rd positive frequency peak (1 mark) *

Your answer

RC filter design

Design a first order RC low-pass filter with a cut-off frequency of 100 Hz
Total: 10 marks

By setting R = 1000 Ohm, calculate the value of C in microFarad (μF), **accurate to *4* significant figures**. (2 marks) *

Your answer

Refer to the PDF document titled "Standard_Capacitor_Values.pdf". Using 3 standard capacitors, what is the closest you can get to the value of C as reported in the previous question? (2 marks) *

Your answer

Create a first order Butterworth low-pass filter in Python with a cut-off frequency of 100 Hz, using scipy.signal.butter. *

Compare the Bode plots of your RC filter and your Butterworth filter, and complete the following (6 marks)

- ☐ The Bode plot magnitude graphs are different
- ☐ The Bode plot phase angle graphs are identical
- ☐ The Bode plot phase angle graphs are different
- ☐ The Bode plot magnitude graphs are identical
- ☐ Other: ____________________

Python filter implementation

Apply your first order Butterworth low-pass filter to your time domain signal, and calculate the FFT of the filtered signal. Compare the peak frequency amplitudes with those you calculated in

Section 2
Total: 8 marks

Enter the amplitude of the 1st positive frequency peak of the filtered signal (**accurate to 3 significant figures**) (1 mark) *

Your answer

Enter the amplitude of the 2nd positive frequency peak of the filtered signal (**accurate to 3 significant figures**) (1 mark) *

Your answer

Enter the amplitude of the 3rd positive frequency peak of the filtered signal (**accurate to 3 significant figures**) (1 mark) *

Your answer

According to you, is the low-pass filter effective in filtering out the 3rd peak frequency? (1 mark) *

- No
- Somewhat
- Yes
- Other: ____

What can be done to increase the effectiveness of the low-pass filter (2 marks) *

- [ ] Nothing, the filter is effective
- [ ] Increase the filter order
- [ ] Reduce the cut-off frequency
- [ ] Other: ____

Compare the phase angles of the frequency components of the filtered signal to that of the unfiltered signal (as reported in Section 2). The differences are caused by (2 marks): *

- Phase angle distortion from using the low-pass filter
- Aliasing
- Random errors characteristic of the FFT algorithm