Problema 1

a) $A = 1v$ $f = 1KHz$ funcion cuadrada

$$f(t) = \begin{cases} A & \text{para } 0 < t < \frac{T}{2} \\ -A & \text{para } \frac{T}{2} < t < T \end{cases}$$

$$a_n = \frac{1}{\pi} \int_0^{\pi} f(t) \cos(nt)\, dt$$

$$= \frac{A}{\pi n} \sin(nt) \Big|_0^{\pi} = \frac{A}{n\pi} \left[\cancel{\sin\left(\frac{n\pi}{2}\right)} \right] \to 0$$

$$b_n = \frac{2}{\pi} \int_0^{\pi/2} f(t) \sin(nt)\, dt$$

$$= -\frac{2A}{n\pi} \cos(nt) \Big|_0^{\pi/} = -\frac{2A}{n\pi} \left[\cos(n\pi) - 1 \right]$$

$$\sum_{n=1}^{\infty} \frac{2A}{n\pi} \left(1 - \cos(n\pi) \right) \sin\left[\left(\frac{n\pi}{T} \right) t \right]$$

cuando n = par la sumatoria es 0 cuando n = impar $1 - \cos(n\pi) = 2$

por lotanto $f(t) = \frac{4A}{\pi} \sum_{n=1}^{\infty} \frac{\sin(nt)}{n}$

los primeros 8 cohepicientes serian:

$$\left[\frac{4}{\pi}, \frac{4}{3\pi}, \frac{4}{5\pi}, \frac{4}{7\pi}, \frac{4}{9\pi}, \frac{4}{11\pi}, \frac{4}{13\pi}, \frac{4}{15\pi}\right]$$

Ingresamos las constantes y la function oscilante en el desarrollo del programa, obteniendo el siguiente codigo y graficacion del mismo de los primeros 8 coeficientes.

```
import matplotlib.pyplot as plt
import math
import numpy

A = 1
F = 1000
T = 1/F

Armonicos = 16

X = numpy.linspace(0, T, 1000)
Y = []
for x in X:
    f_sum = 0
    for n in range(1, Armonicos + 1):
        if n%2 == 1:
            f_sum += 4*(math.sin(2*math.pi*n*F*x))/(n*math.pi)
        else:
            pass
    Y.append(f_sum)

plt.plot(X, Y)
plt.show()
```

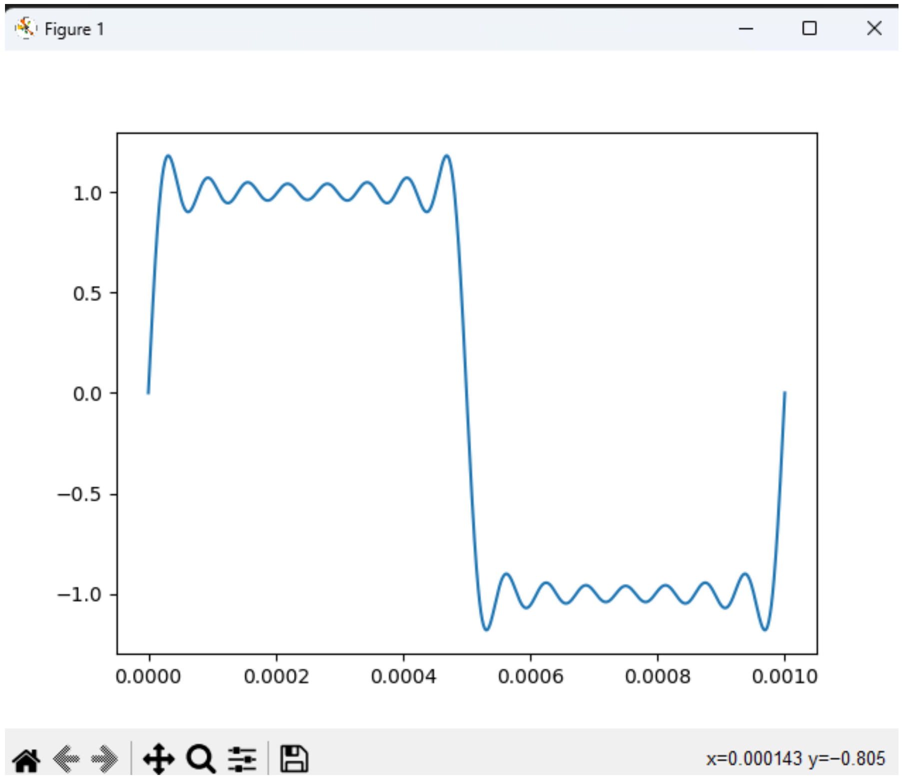
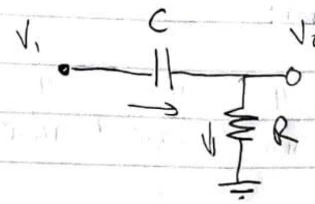


Figure 1
1.0
0.5
0.0
−0.5
−1.0
0.0000
0.0002
0.0004
0.0006
0.0008
0.0010
x=0.000143 y=−0.805

Problema 2



c)

$$i_C = i_R$$

$$\frac{V_0 - V_1}{X_C} = \frac{0 - V_0}{R}$$

$$-\frac{V_1}{X_C} = -V_0\left(\frac{1}{X_C} + \frac{1}{R}\right)$$

$$\frac{V_0}{V_1} = \frac{1}{X_C\left(\frac{1}{X_C} + \frac{1}{R}\right)}$$

$$\boxed{F(j\omega) = \frac{R}{X_C + R} \qquad X_C = -\frac{j}{\omega C}}$$

d) $f_0 = 10\,KHz$

$$f_0 = \frac{1}{2\pi RC}$$

$$C = \frac{1}{2\pi(10KHz)R}$$

$\forall R$

Cualquier valor que satisfaga esta igualdad

Valores Comerciales de Capacitores

pF	nF	µF	mF
1	1	1	1
1.2	1.2	1.2	1.2
1.5	1.5	1.5	1.5
1.8	1.8	1.8	1.8
2.2	2.2	2.2	2.2
2.7	2.7	2.7	2.7
3.3	3.3	3.3	3.3
3.9	3.9	3.9	3.9
4.7	4.7	4.7	4.7
5.6	5.6	5.6	5.6
6.8	6.8	6.8	6.8
8.2	8.2	8.2	8.2
10	10	10	10
12	12	12	12
15	15	15	15
18	18	18	18
22	22	22	22
27	27	27	27
33	33	33	33
39	39	39	39
47	47	47	47
56	56	56	56
68	68	68	68
82	82	82	82
100	100	100	
120	120	120	
150	150	150	
180	180	180	
220	220	220	
270	270	270	
330	330	330	
390	390	390	
470	470	470	
560	560	560	
680	680	680	
820	820	820	

**Los valores de las resistencias existen en múltiplos de 0.01, 0.1, 1, 10, 100, 1000, 10000 y 100000 llegando como máximo a 10 MΩ

e) Dados todos los valores comerciales tanto de resistencias como de capacitors en las siguientes tablas:

VALORES COMERCIALES DE RESISTENCIAS						
1Ω	10 Ω	100 Ω	1 KΩ	10 KΩ	100 KΩ	1 MΩ
1,2 Ω	12 Ω	120 Ω	1,2 KΩ	12 KΩ	120 KΩ	1,2 MΩ
1,5 Ω	15 Ω	150 Ω	1,5 KΩ	15 KΩ	150 KΩ	1,5 MΩ
1,8 Ω	18 Ω	180 Ω	1,8 KΩ	18 KΩ	180 KΩ	1,8 MΩ
2,2 Ω	22 Ω	220 Ω	2,2 KΩ	22 KΩ	220 KΩ	2,2 MΩ
2,7 Ω	27 Ω	270 Ω	2,7 KΩ	27 KΩ	270 KΩ	2,7 MΩ
3,3 Ω	33 Ω	330 Ω	3,3 KΩ	33 KΩ	330 KΩ	3,3 MΩ
3,9 Ω	39 Ω	390 Ω	3,9 KΩ	39 KΩ	390 KΩ	3,9 MΩ
4,7 Ω	47 Ω	470 Ω	4,7 KΩ	47 KΩ	470 KΩ	4,7 MΩ
5,1 Ω	51 Ω	510 Ω	5,1 KΩ	51 KΩ	510 KΩ	5,1 MΩ
5,6 Ω	56 Ω	560 Ω	5,6 KΩ	56 KΩ	560 KΩ	5,6 MΩ
6,8 Ω	68 Ω	680 Ω	6,8 KΩ	68 KΩ	680 KΩ	6,8 MΩ
8,2 Ω	82 Ω	820 Ω	8,2 KΩ	82 KΩ	820 KΩ	8,2 MΩ
SERVICIO TECNICO DEL MILAGRO						10 MΩ

Se creo un script de python para poder obtener todos los valores comerciones que satisfagan la ecaucion de la fecuencia de corte y con un error absoluto de mas-menos a 300 hertz, se obtuvieron los siguientes valores:

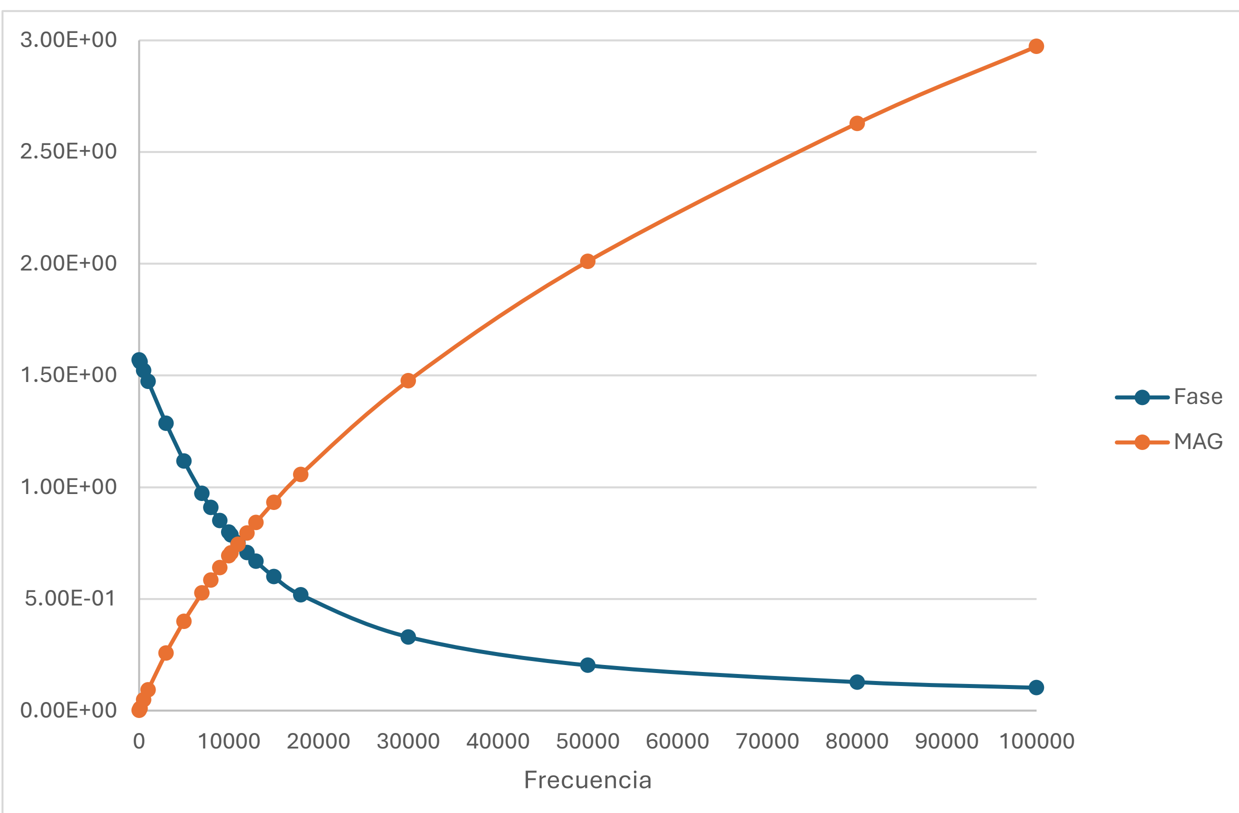
Frec de Corte	Res	Cap
10261.44	3.3	4.7e-06
10261.44	4.7	3.2999999999999997e-06
10261.44	33	4.7000000000000005e-07
10261.44	47	3.3e-07
10261.44	330	4.7000000000000004e-08
10261.44	470	3.3000000000000004e-08
10261.44	3300.0	4.700000000000001e-09
10261.44	4700.0	3.3e-09
10261.44	33000	4.7e-10
10261.44	47000	3.3e-10
10261.44	330000	4.7e-11
10261.44	470000	3.3e-11
10261.44	3300000.0	4.7e-12
10261.44	4700000.0	3.2999999999999997e-12

f (Hz)	Mag	Fase
10	9.74E-04	1.57E+00
100	9.70E-03	1.56E+00
500	4.76E-02	1.52E+00
1000	9.30E-02	1.47E+00
3000	2.57E-01	1.29E+00
5000	4.00E-01	1.12E+00
7000	5.26E-01	9.72E-01
8000	5.84E-01	9.09E-01
9000	6.40E-01	8.51E-01
10000	6.94E-01	7.98E-01
10240	7.06E-01	7.86E-01
11000	7.45E-01	7.51E-01
12000	7.94E-01	7.07E-01
13000	8.41E-01	6.68E-01
15000	9.32E-01	6.00E-01
18000	1.06E+00	5.18E-01
30000	1.48E+00	3.30E-01
50000	2.01E+00	2.02E-01
80000	2.63E+00	1.28E-01
100000	2.97E+00	1.02E-01

$$Fase = \frac{\pi}{2} - \arctan(2\pi f RC)$$

$$\|G\| = \frac{2\pi f RC}{\sqrt{1 + 2\pi f RC}}$$

R = 3300 C = 4.7E-9



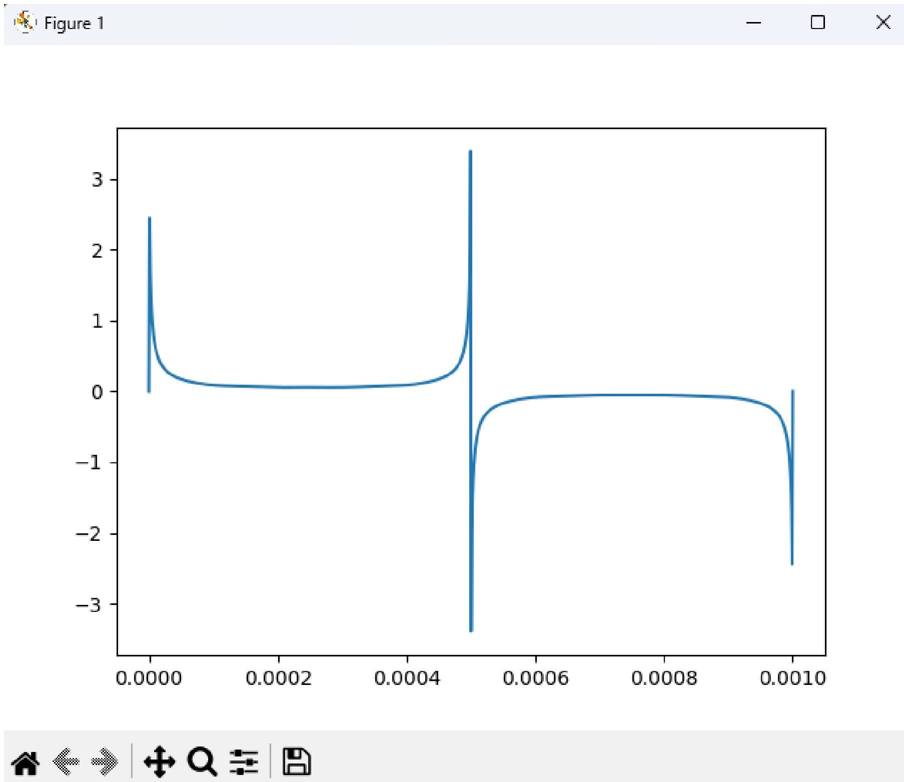
Problema 3

Se agrego en el codigo del inciso a) la variable ganancia el cual se anexo la ecuacion que obtuvimos del problema dos llamada MAG, esto se vera afectado por la frecuencia de cada armónico, el coigo quedo de la siguiente manera:

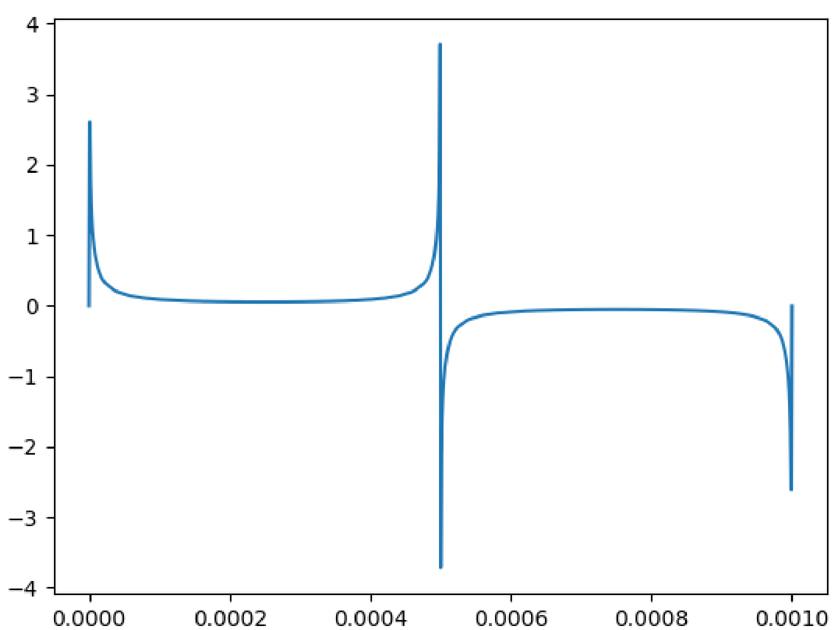
```
import matplotlib.pyplot as plt
import math
import numpy

A = 1
F = 1000
T = 1/F
R = 3300 #ohmns
C = .000000004 #faradios

Armonicos = 20000

X = numpy.linspace(0, T, 1000)
Y = []
for x in X:
    f_sum = 0
    for n in range(1, Armonicos + 1):
        if n%2 == 1:
            f = n*F
            Ganancia = 2*math.pi*f*R*C/math.sqrt(1+2*math.pi*f*R*C)
            f_sum += Ganancia*4*(math.sin(2*math.pi*f*x))/(n*math.pi)
        else:
            pass
    Y.append(f_sum)

plt.plot(X, Y)
plt.show()
```

Como salida obtuvimos:

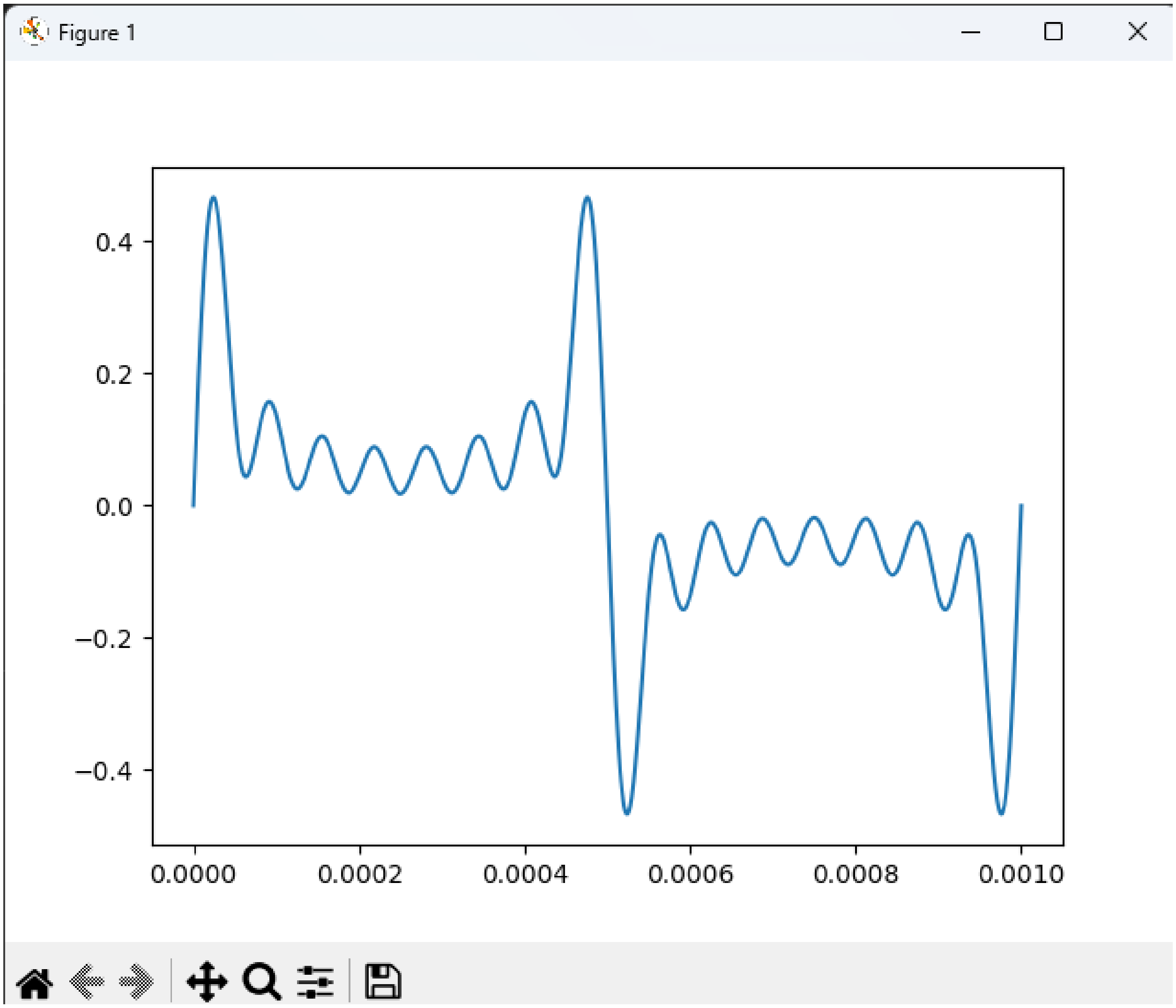
para frecuencia maxima de 20kHz



Para frecuencia maxima de 50kHz



Para frecuancia maxima de 16Hz:



Conclusion:

Los picos que se pueden observer en cada grafico son cuando la onda cuadrada sufre un cambio de estado, esto es porque alli las frecuencias altas son las que predominan, y se dejan pasar gracias al fitrado pasa altas.