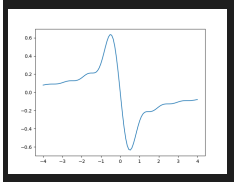
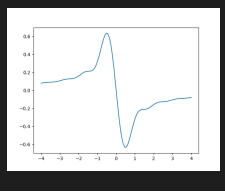
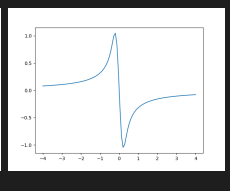
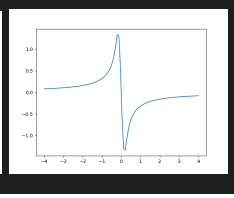
Aarnav Devulapalli

Question 1

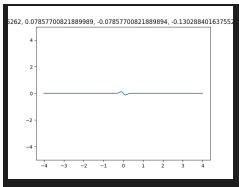
Part a

```
import numpy as np
import matplotlib.pyplot as plt
import math
t = [0,0.001,0.5,1]
s = np.linspace(-4,4,100)
u = []
u = []
i = []
t_values = [0, 0.001, 0.5, 1]
for t in t_values:
  for j in range(100):
    s_j = s[j]
    term1 = 1 / (np.pi * 1j * s_j)
    term2 = -2j * np.sin(2 * np.pi * s_j) / ((2 * np.pi * 1j * s_j) ** 2)
    term3 = math.exp((s_j * s_j * -4 * math.pi * math.pi * t))
    u_j = term1 + term2 * term3
    u.append(u_j)
  i = [ele.imag for ele in u]
  plt.plot(s, i)
  plt.show()
  u = []
  i = []
```

Part b

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
t = np.linspace(0, 2, 21)
s = np.linspace(-4, 4, 100)
fig, ax = plt.subplots()
def animate(i):
  u = []
  for k in range(100):
    u.append((1/(np.pi*1j*s[k])-2j*np.sin(2*np.pi*s[k])/((2*np.pi*1j*s[k])**2))*math.exp((s[k]*s[k]*-4*math.pi*math.pi*t[i])))
  i = [ele.imag for ele in u]
  ax.clear()
  ax.plot(s, i)
  ax.set_ylim(-5, 5)
  ax.set_title('Animation frame {}'.format(i))
ani = FuncAnimation(fig, animate, frames=len(t), interval=200)
plt.show()
```



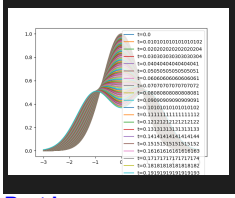
Part c

Part d

Question 2

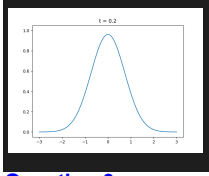
Part a

```
import numpy as np
import matplotlib.pyplot as plt
def phi(x):
  return np.exp(-x**2)
x = np.linspace(-3, 3, 1000)
t = np.linspace(0, 1, 100)
c = 1
def u(x, t):
  return 0.5*(phi(x-c*t) + phi(x+c*t))
for i in range(len(t)):
  plt.plot(x, u(x, t[i]), label='t='+str(t[i]))
plt.legend()
plt.show()
```



Part b

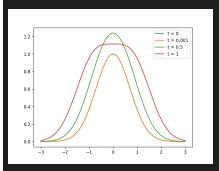
```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
def phi(x):
  return np.exp(-x**2)
x = np.linspace(-3, 3, 1000)
t = np.arange(0, 2.1, 0.1)
c = 1
def u(x, t):
  return 0.5*(phi(x-c*t) + phi(x+c*t))
fig, ax = plt.subplots()
line, = ax.plot(x, u(x, 0))
def update(frame):
  line.set_ydata(u(x, t[frame]))
  ax.set_title('t = {:.1f}'.format(t[frame]))
  return line,
anim = FuncAnimation(fig, update, frames=len(t), interval=100)
plt.show()
```



Question 3

Part a

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import quad
c = 1
x_range = np.linspace(-3, 3, 1000)
t_values = [0, 0.001, 0.5, 1]
def phi(x):
  return np.exp(-x**2)
def u(x, t):
  integral = lambda s: phi(s)*np.heaviside(t - np.abs(x - s) / c, 0.5)
  integral_value, _ = quad(integral, x - c*t, x + c*t)
  return (phi(x + c*t) + phi(x - c*t))/2 + (1/(2*c))*integral_value
for t in t_values:
  u_values = [u(x, t) for x in x_range]
  plt.plot(x_range, u_values, label=f"t = {t}")
plt.legend()
plt.show()
```



Part b

```
from matplotlib.animation import FuncAnimation
t_range = np.arange(0, 2, 0.1)
def update_plot(frame):
  plt.cla()
  u_values = [u(x, frame) for x in x_range]
  plt.plot(x_range, u_values)
  plt.ylim([-1, 1])
  plt.title(f"t = {frame:.1f}")
fig = plt.figure()
animation = FuncAnimation(fig, update_plot, frames=t_range, blit=False)
plt.show()
```