Question 1-a

$$k(x, x') = x^T x'$$

$$k(x', x) = x'^T x$$

k is symmetric

$$v^t K v = v^T X^T X v = (Xv)^T (Xv) = \ L_2 \text{ norm}$$

K is PSD

Question 1-b

$$k(x, x') = f(x) f(x')$$

$$k(x', x) = f(x') f(x)$$

k is symmetric

$$K_{ij} = f(x_i) f(x_j) = u_i u_j = u u^T$$

$$v^t K v = v^T u u^T v = (u^T v)^T (u^T v) = (u^T v)^2 \geq 0$$

K is PSD

Question 1-c

$$k(x, x') = k_1(x, x') + k_2(x, x') = k_1(x', x) + k_2(x', x) = k(x', x)$$

K is symmetric

$$v^T K^{(1)} v \geq 0, \quad v^T K^{(2)} v \geq 0$$

$$v^T K^{(1)} v + v^T K^{(2)} v \geq 0$$

K is PSD

Question 2-a

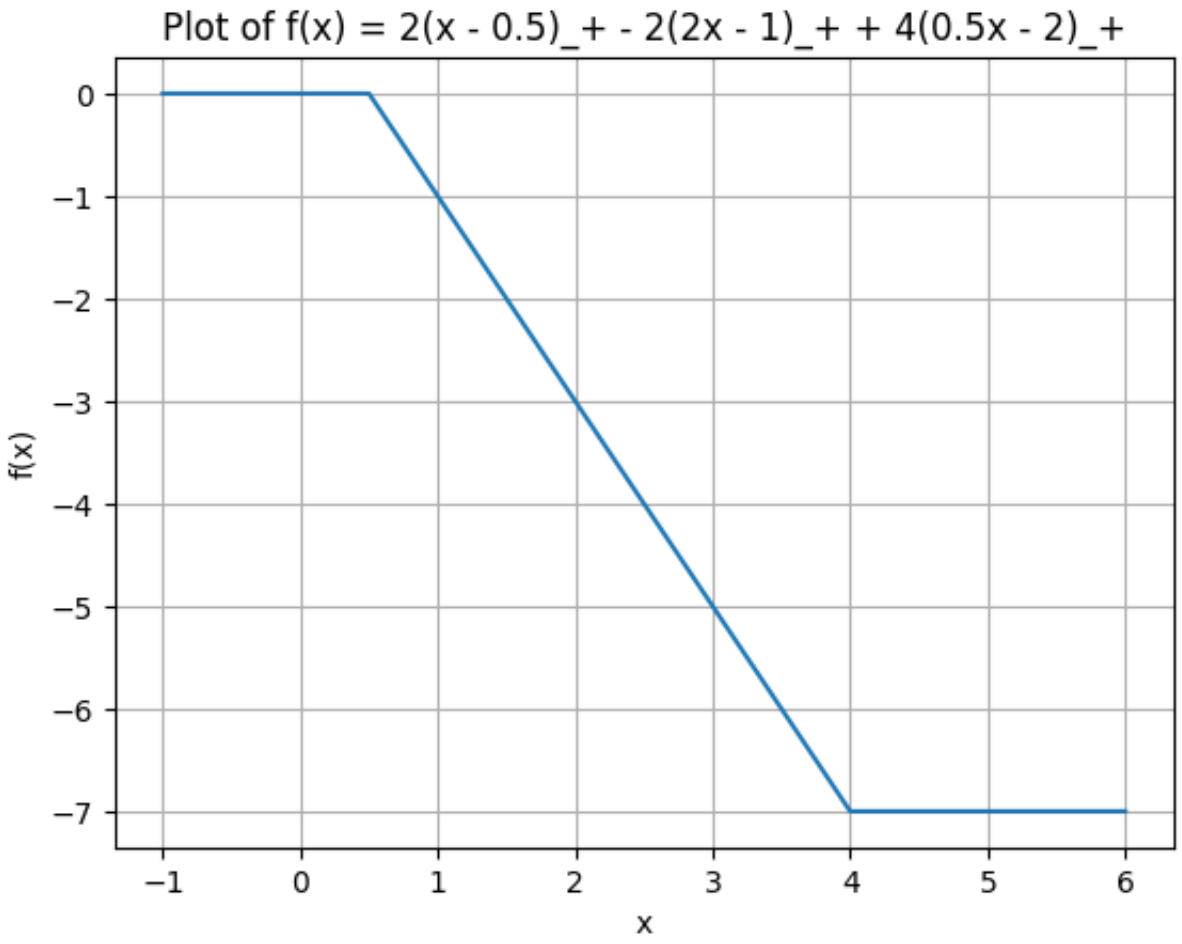
In [1]:
```
import numpy as np
import matplotlib.pyplot as plt

# Define ReLU
def relu(z):
    return np.maximum(0, z)
```

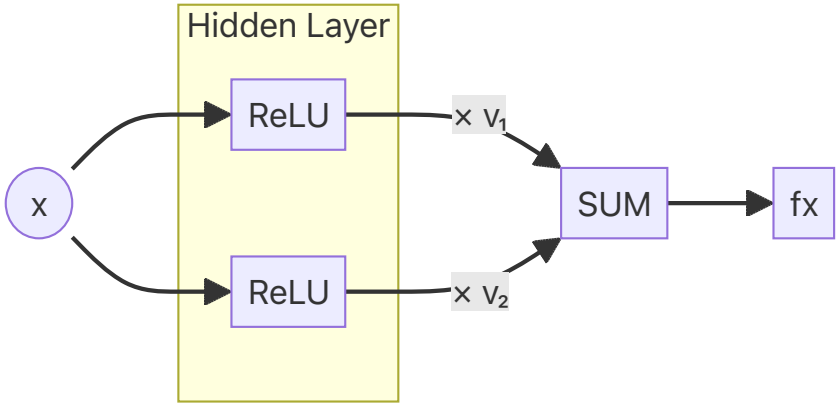
```python
# Define the piecewise function f(x)
def f(x):
    return 2 * relu(x - 0.5) - 2 * relu(2 * x - 1) + 4 * relu(0.5 * x - 2)

# Generate x values and compute f(x)
x = np.linspace(-1, 6, 1000)
y = f(x)

# Plot
plt.figure()
plt.plot(x, y)
plt.xlabel('x')
plt.ylabel('f(x)')
plt.title('Plot of f(x) = 2(x - 0.5)_+ - 2(2x - 1)_+ + 4(0.5x - 2)_+')
plt.grid(True)
plt.show()
```



Question 2-b



Question 2-c

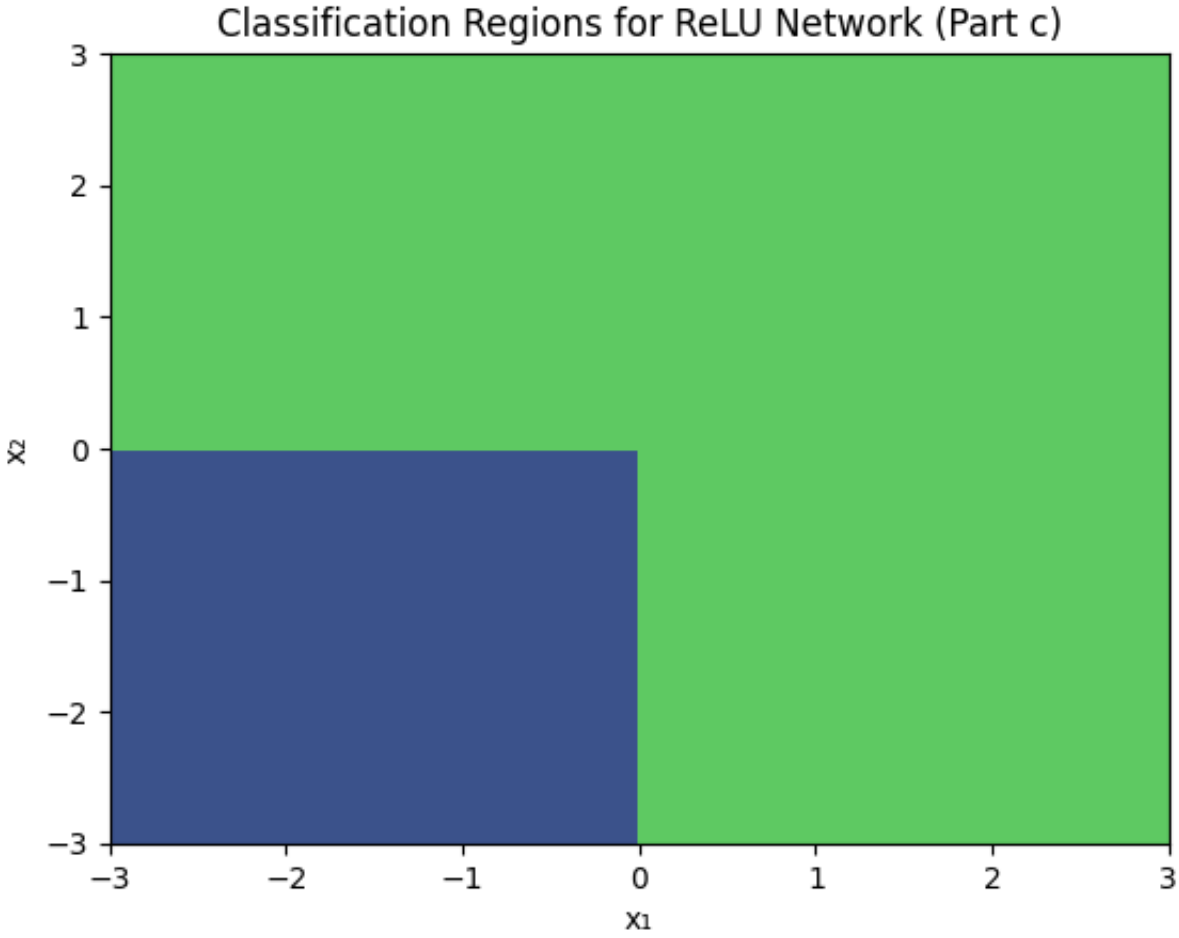
Neuron	(w_j)	(b_j)	output weight (v_j)
1	([1,0])	(0)	(+1)
2	([0,1])	(0)	(+1)

In [2]:
```python
import numpy as np
import matplotlib.pyplot as plt

# Define the classifier function
def predict_label(x1, x2):
    return np.sign(np.maximum(0, x1) + np.maximum(0, x2))

# Create a grid over which to evaluate
x = np.linspace(-3, 3, 400)
y = np.linspace(-3, 3, 400)
xx, yy = np.meshgrid(x, y)
Z = predict_label(xx, yy)

# Plot the classification regions
plt.figure()
plt.contourf(xx, yy, Z, levels=[-1, 0, 1])
plt.xlabel('x₁')
plt.ylabel('x₂')
plt.title('Classification Regions for ReLU Network (Part c)')
plt.show()
```


Classification Regions for ReLU Network (Part c)
3
2
1
0
−1
−2
−3
x2
−3
−2
−1
0
1
2
3
x1