Section 1:

Question:
Reproduce the steps below to understand how the sigmoid (logistic) function is implemented and used in logistic regression. Then answer the following:

1. **"Write a function to compute the sigmoid activation using NumPy (vectorized version)."**
2. **"Compute the value of $z = w^T x + b$ for the given vectors:**
   $x = [1, 2, 3]$, $w = [0.5, 0.3, 0.2]$, $b = 0.4$."
3. **"Compute the activation $a = \text{sigmoid}(z)$."**
4. **"Extend the computation to a matrix $X = [[1, 2, 3], [1.5, 3.1, 4], [5, 6, 3]]^T$ and calculate $Z = w^T X + b$, then $A = \text{sigmoid}(Z)$."**
5. **"Given $Y = [[1, 0, 1]]$, compute both:"**
   - **"The Least Squares cost"**
   - **"The Log Loss (cross-entropy) cost"**

Section 2:

Using the previous definitions and results:

1. Write the initial parameters:

$$w = \begin{bmatrix} 0.5 \\ 0.3 \\ 0.2 \end{bmatrix}, \quad b = 0.4$$

Explain what these parameters represent in a logistic regression model.

2. Compute the gradients of the cost function with respect to w and b:

$$\Delta w = \frac{1}{m} X (A - Y)^T, \quad \Delta b = \frac{1}{m} \sum (A - Y)$$

Then, using a learning rate $\alpha = 0.01$, update the parameters:

$$w := w - \alpha \Delta w, \quad b := b - \alpha \Delta b$$

What are the new values of w and b?

3. Implement two functions:

- `forward(w, b, X, Y)` to compute predictions and cost.
- `backward(w, b, X, Y, A)` to compute gradients and update parameters.

4. Initialize w randomly, $b = 0$, and run 1000 iterations of gradient descent using the two functions above. Store and plot the evolution of the cost function.