Pseudocode

```
Algorithm: RidgeRegression

Input:
    X  (training data: n × d)
    y  (target values: n)
    α  (learning rate)
    λ  (regularization parameter)
    T  (number of iterations)

Output:
    w  (weight vector)
    b  (bias)

Initialize w = zeros(d)
Initialize b = 0

for i = 1 to T do
    y_hat = X · w + b
    error = y_hat − y

    dw = (1/n) · (Xᵀ · error + λ · w)
    db = (1/n) · sum(error)

    w = w − α · dw
    b = b − α · db
end for

return w, b
```

Python Code

```python
import numpy as np

def ridge_regression(X, y, alpha=0.01, lambda_=0.1, iterations=1000):
    n, d = X.shape
    w = np.zeros(d)
    b = 0

    for _ in range(iterations):
        y_hat = np.dot(X, w) + b
```

```
        error = y_hat - y

        dw = (1 / n) * (np.dot(X.T, error) + lambda_ * w)
        db = (1 / n) * np.sum(error)

        w = w - alpha * dw
        b = b - alpha * db

    return w, b
```