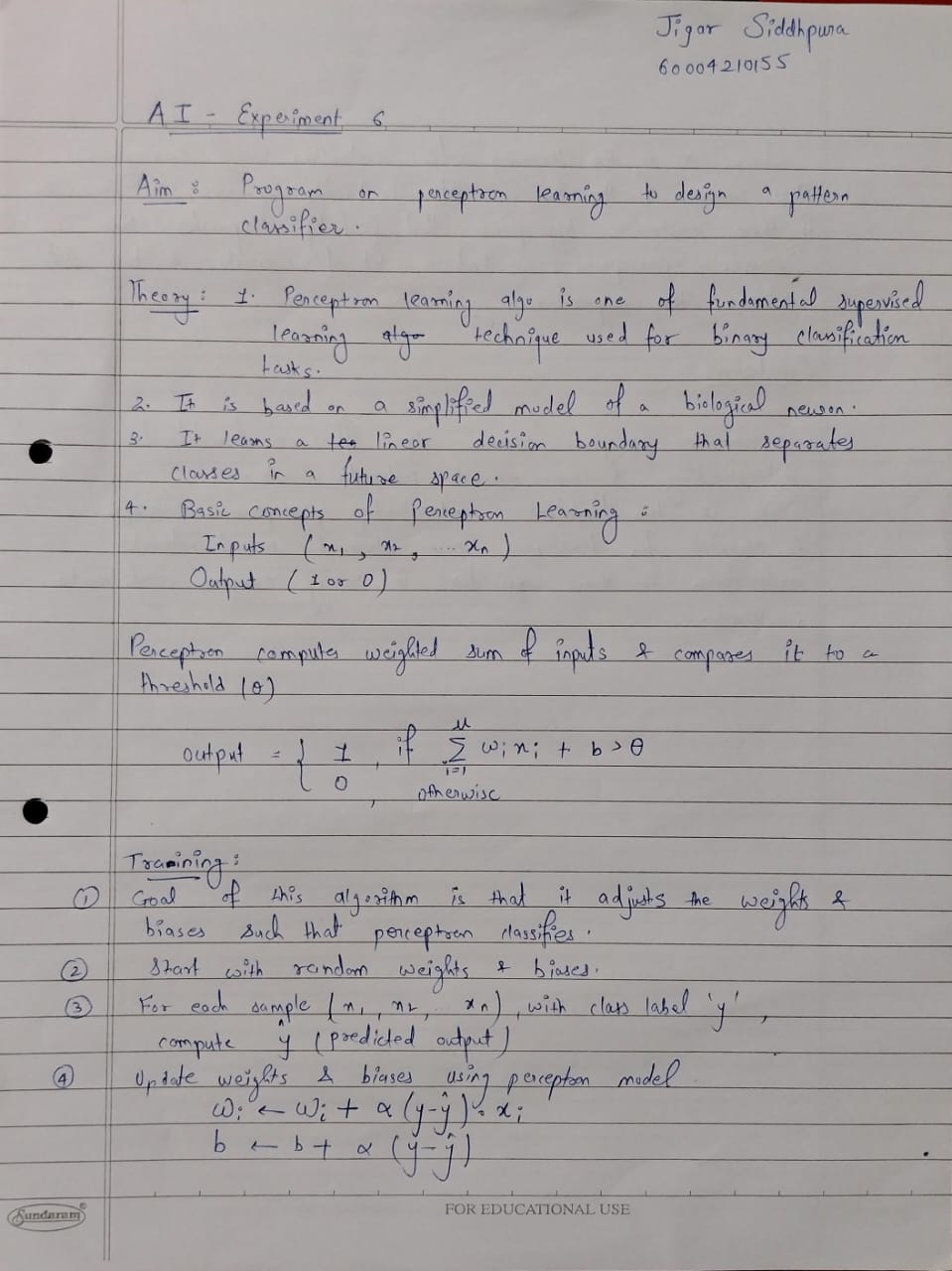
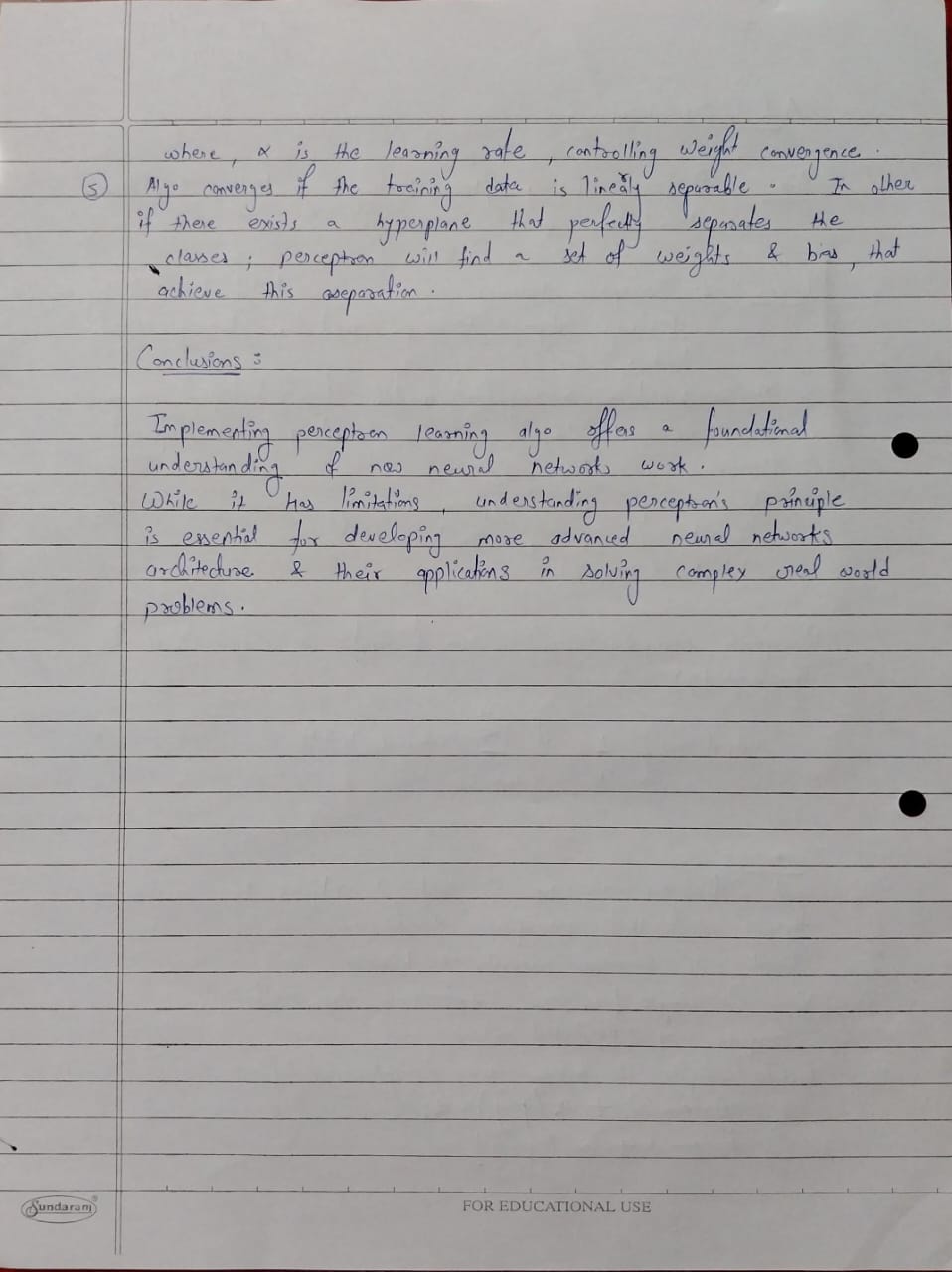
**Name: Jigar Siddhpura SAPID:** 60004200155

**DIV: C/C2** **Branch:** Computer Engineering

AI EXPERIMENT 6 - Perceptron Learning

**Theory:**

****



**Code :**

import numpy as np

*# Input features*

X = np.array([

    [1, 0, 1, 0, 0, 0, 1, 1, 1],

    [1, 0, 0, 1, 0, 0, 1, 1, 1],

    [1, 1, 0, 1, 0, 0, 1, 1, 1],

    [1, 0, 0, 1, 0, 0, 1, 1, 1],

    [1, 0, 0, 1, 0, 1, 1, 1, 1],

    [1, 0, 1, 0, 1, 1, 1, 0, 1],

    [1, 0, 1, 1, 1, 0, 1, 0, 1],

    [1, 0, 1, 1, 0, 1, 1, 0, 1],

    [1, 0, 0, 1, 1, 1, 1, 1, 1],

    [1, 1, 1, 1, 1, 1, 1, 0, 1]

])

*# Initial weights*

W = np.array([-3,1,-2,0.7,-1,1,0,3,2])

*# Desired outputs*

d = np.array([1,0,1,0,1,0,1,0,1,0])

c = 1 *# Learning rate*

epochs = 6 *# Number of epochs*

*# Training the perceptron*

for epoch in range(epochs):

    print("Iteration ", epoch + 1)

    for i in range(len(X)):

*# Compute the net input*

        net = np.dot(X[i], W)

*# Apply the step function*

        op = 1 if net > 0 else 0

*# Compute the error*

        error = d[i] - op

*# Update weights*

        dW = c \* error \* X[i]

        W += dW

        print("W", i, W)

print("\nW after ", epoch + 1, " epochs ", W)

print("Final W after ", epochs, "epochs:")

print(W)

*# Testing the perceptron with a new input*

test\_input1 = [1, 0, 1, 1, 0, 0, 1, 1, 0]

test\_input2 = [1, 0, 0, 1, 1, -1, 1, 1, 1]

def test(test\_data,weights):

net = np.dot(test\_data, weights)

*# Apply the step function to get the output*

    output = 1 if net > 0 else 0

    return output

print(f"Output for test input {test\_input1}:", test(test\_input1,W))

print(f"Output for test input {test\_input2}:", test(test\_input2,W))

**Output :**

