

## Experiment 1

Harshvardhan Singh  
1019161

Source Code:

```
exp1.py > ...
1  '''
2  Implement Perceptron algorithm to simulate AND logic gate
3  '''
4  import numpy as np
5  def unitStep(v):
6      if v >= 0:
7          return 1
8      else:
9          return 0
10 def perceptronModel(x, w, b):
11     v = np.dot(w, x) + b
12     y = unitStep(v)
13     return y
14 def AND_logicFunction(x):
15     w = np.array([1, 1])
16     b = -1.5
17     return perceptronModel(x, w, b)
18 test1 = np.array([0, 1])
19 test2 = np.array([1, 1])
20 test3 = np.array([0, 0])
21 test4 = np.array([1, 0])
22 print("AND({}, {}) = {}".format(0, 1, AND_logicFunction(test1)))
23 print("AND({}, {}) = {}".format(1, 1, AND_logicFunction(test2)))
24 print("AND({}, {}) = {}".format(0, 0, AND_logicFunction(test3)))
25 print("AND({}, {}) = {}".format(1, 0, AND_logicFunction(test4)))
```

Output

```
PS C:\Users\admin\Desktop\DC Lab>
y"
AND(0, 1) = 0
AND(1, 1) = 1
AND(0, 0) = 0
AND(1, 0) = 0
PS C:\Users\admin\Desktop\DC Lab>
```