|  |  |
| --- | --- |
|  | **DEPARTMENT OF COMPUTER ENGINEERING** |

Experiment No. 01

|  |  |
| --- | --- |
| Semester | B.E. Semester VIII – Computer Engineering |
| Subject | Deep Learning Lab |
| Subject Professor In-charge | Prof. Kavita Shirsat |
| Academic Year | 2024-25 |

|  |  |
| --- | --- |
| Student Name | Deep Salunkhe |
| Roll Number | 21102A0014 |

**Title:** Implementation of McCulloch-Pitts algorithm

**Explanation:**

The **McCulloch-Pitts (M-P) Model**, introduced in 1943 by Warren McCulloch and Walter Pitts, is one of the earliest theoretical models of artificial neurons. It provides a foundation for modern artificial neural networks and mimics basic neural behavior in the human brain.

**Implementation:**

**import** **java.util.\***;

**public** **class** MP {

**private** **static** **Scanner** sc **=** **new** Scanner(System.in);

**private** **int** gofx(**int**[] x) {

**int** result **=** 0;

**for** (**int** value **:** x) {

            result **+=** value;

        }

**return** result;

    }

**private** **int** or(**int** gofx) {

**return** gofx **>** 0 **?** 1 **:** 0;

    }

**private** **int** AND(**int** gofx, **int** n) {

**return** gofx **==** n **?** 1 **:** 0;

    }

**private** **int** NAND(**int** gofx, **int** n) {

**return** gofx **==** n **?** 0 **:** 1;

    }

**private** **int** NOR(**int** gofx) {

**return** gofx **==** 0 **?** 1 **:** 0;

    }

**private** **void** generateTruthTable(**int** n) {

**int** rows **=** (**int**) Math.pow(2, n);

**int**[][] table **=** **new** **int**[rows][n];

        // Generate binary combinations for the truth table

**for** (**int** i **=** 0; i **<** rows; i**++**) {

**for** (**int** j **=** 0; j **<** n; j**++**) {

                table[i][j] **=** (i **>>** (n **-** j **-** 1)) **&** 1;

            }

        }

        System.out.println("Truth Table for " **+** n **+** " inputs:");

        System.out.println("Inputs" **+** "\tAND\tOR\tNAND\tNOR");

        // Evaluate gates for each row

**for** (**int**[] row **:** table) {

**int** gofx **=** gofx(row);

**int** andResult **=** AND(gofx, n);

**int** orResult **=** or(gofx);

**int** nandResult **=** NAND(gofx, n);

**int** norResult **=** NOR(gofx);

            // Print the row and results

**for** (**int** value **:** row) {

                System.out.print(value **+** " ");

            }

            System.out.println("\t" **+** andResult **+** "\t" **+** orResult **+** "\t" **+** nandResult **+** "\t" **+** norResult);

        }

    }

**public** **static** **void** main(**String**[] args) {

        System.out.println("Enter the number of inputs: ");

**int** n **=** sc.nextInt();

**MP** mp **=** **new** MP();

        mp.generateTruthTable(n);

    }

}

**Output:**

