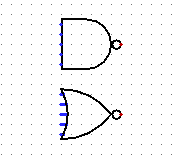
**Q1:** Which gates are categorized as universal gates and how are they used?

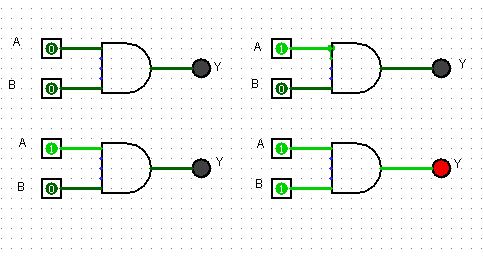
**Ans:** The OR, AND, and NOT are the three basic logic gates as they together can construct the logic circuit for any given Boolean expression. NOR and NAND gates have the property that they individually can be used to hardware-implement any logic circuit. For this reason, the NAND and NOR gates are called universal gates.

That’s how NAND and NOR gates look like:

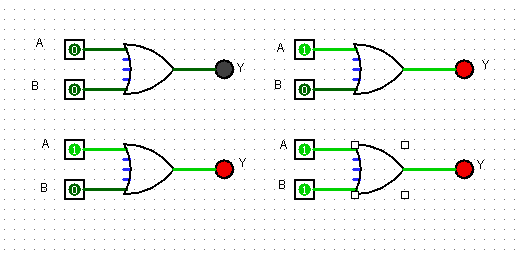
1-NAND

2-NOR

**Q2:** Verify the truth table of AND gate and OR gate.

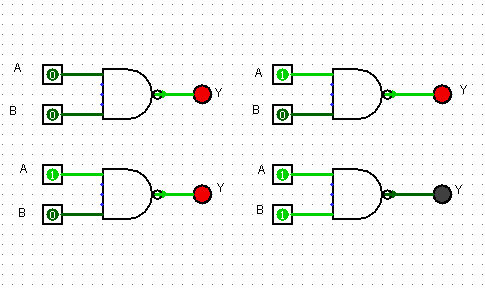
 **AND gate:**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **0** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |
| **1** | **1** | **1** |

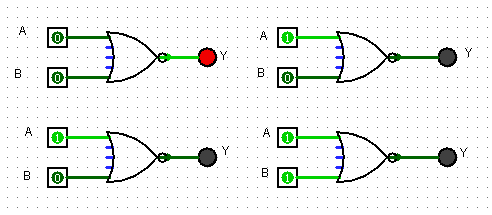
**OR gate:**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **0** |
| **0** | **1** | **1** |
| **1** | **0** | **1** |
| **1** | **1** | **1** |

**Q3:** Verify the truth table of NOR and NAND gate.

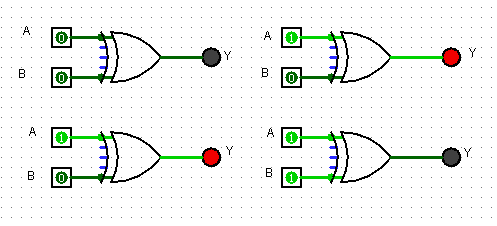
**NAND gate:**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **1** |
| **0** | **1** | **1** |
| **1** | **0** | **1** |
| **1** | **1** | **0** |

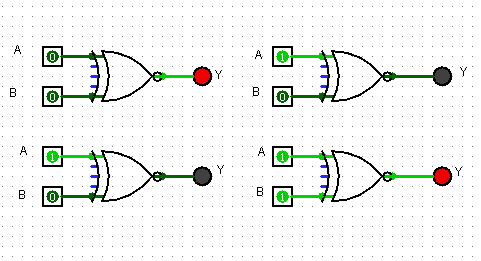
**NOR gate:**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **1** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |
| **1** | **1** | **0** |

**Q4:** Verify the truth table of XOR gate and XNOR gate.

**XOR gate:**

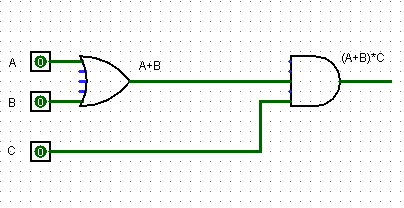
|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **0** |
| **0** | **1** | **1** |
| **1** | **0** | **1** |
| **1** | **1** | **0** |

**XNOR gate:**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Y** |
| **0** | **0** | **1** |
| **0** | **1** | **0** |
| **1** | **0** | **0** |
| **1** | **1** | **1** |

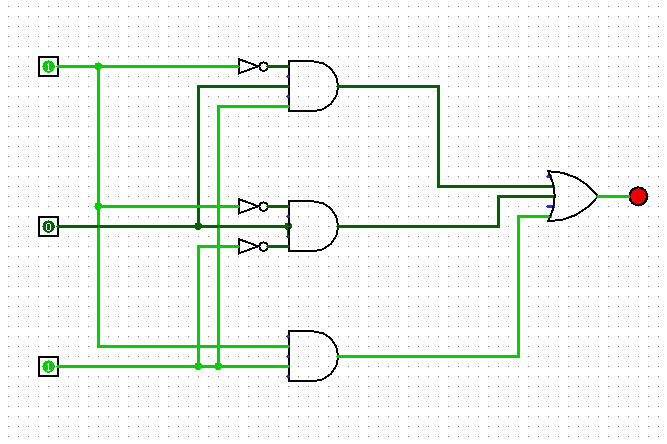
**Q5:** Convert the following logic gate into a Boolean expression writing Boolean sub expression next to each gate output in the diagram.

**Ans:** (A+B) \*C

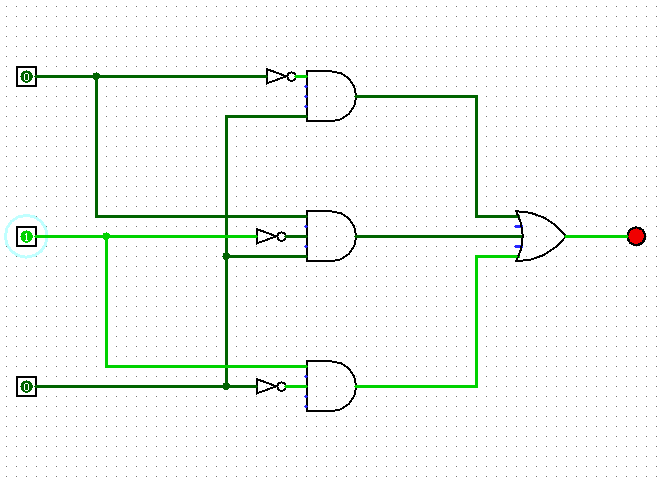
****

**Q6: Draw the following function in circuit maker.**

**1/ F=X’YZ=X’YZ’+XZ**

****

**2/**

****