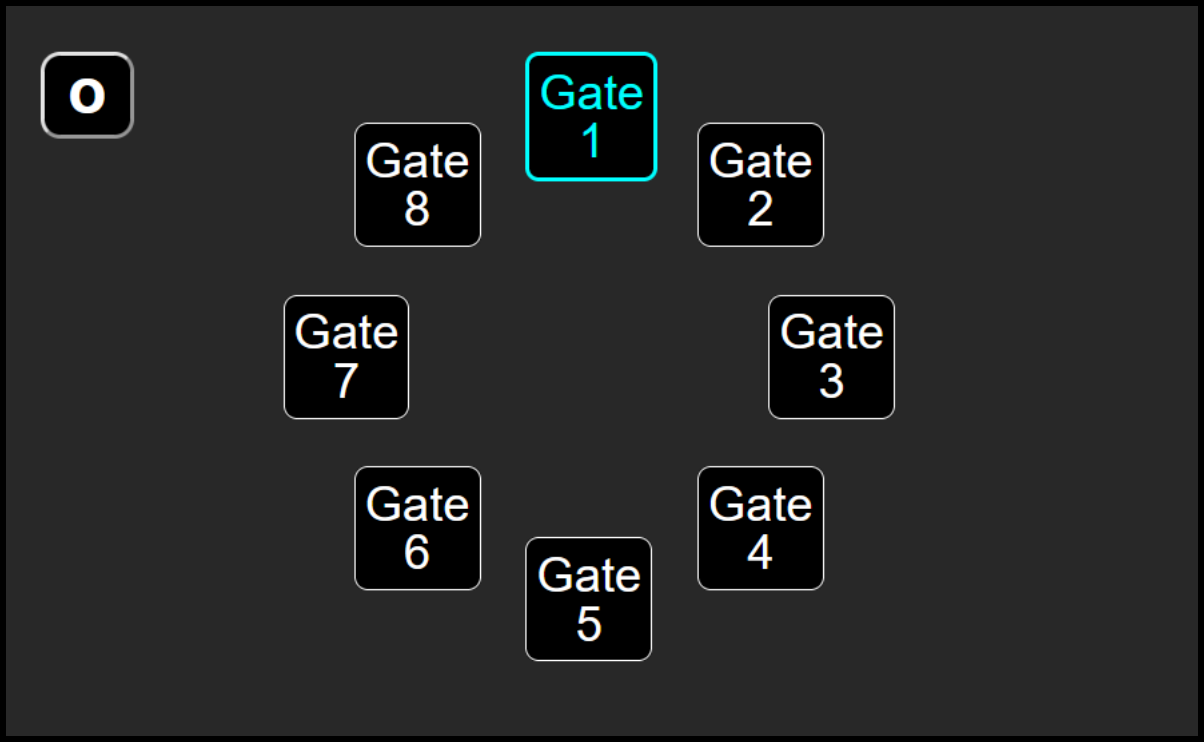
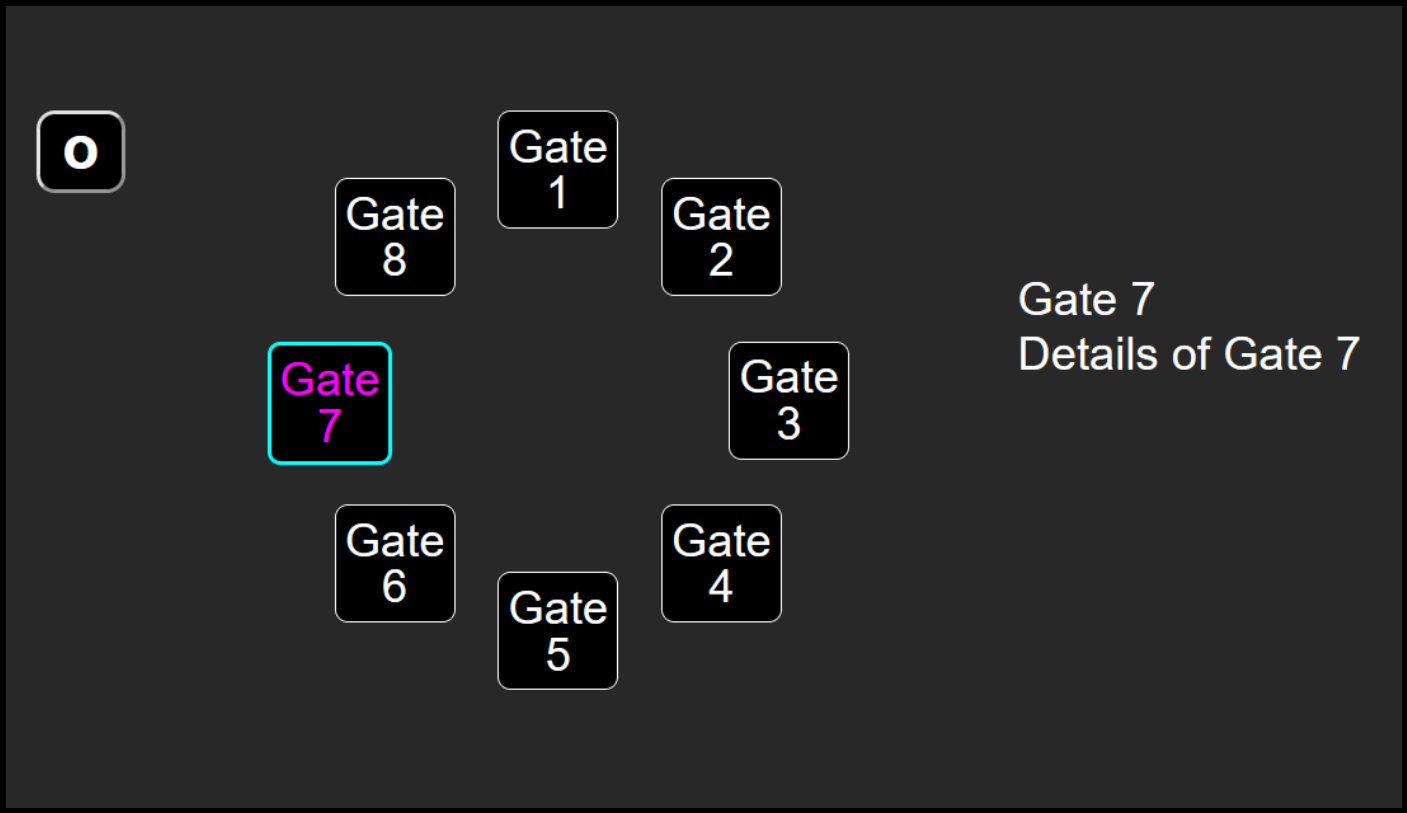
Topalian JavaScript Trigonometry Pie Menu

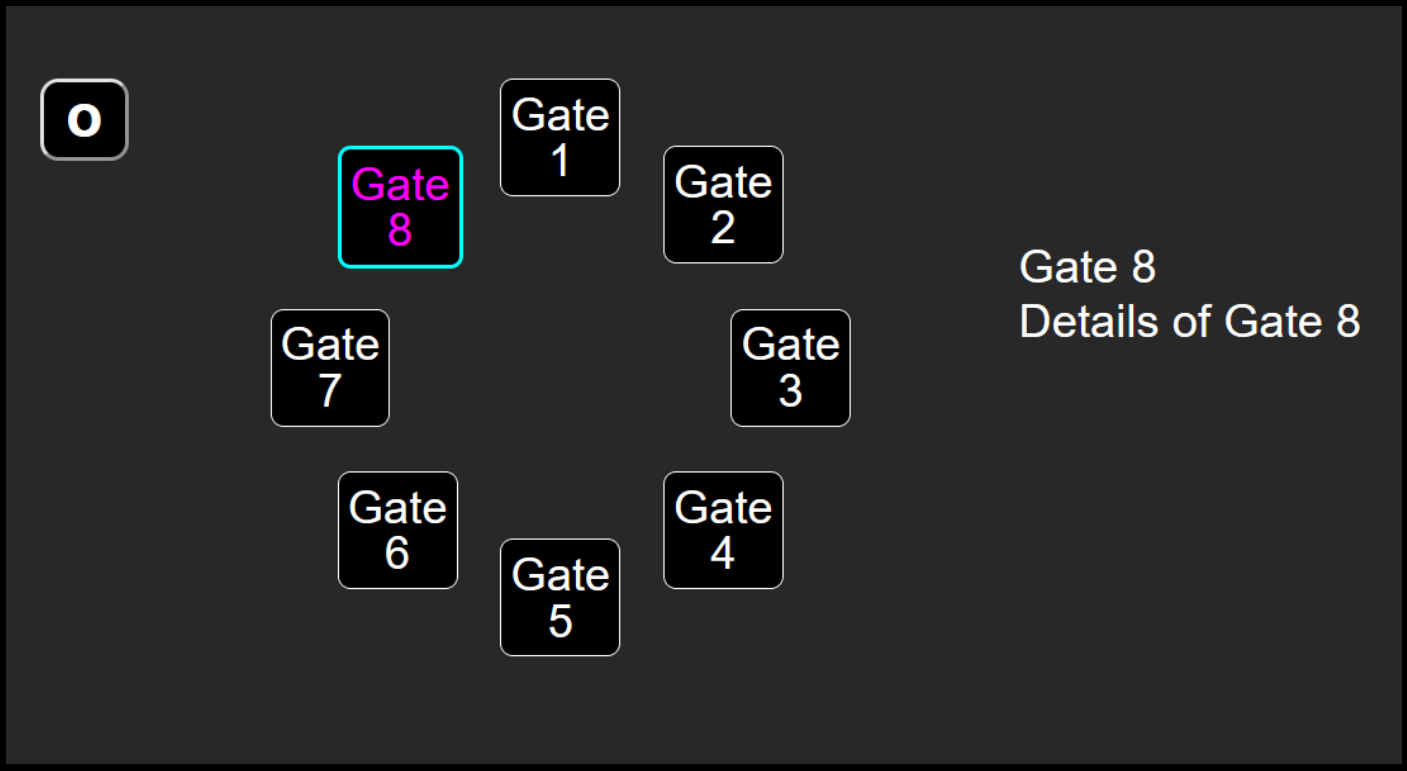
by

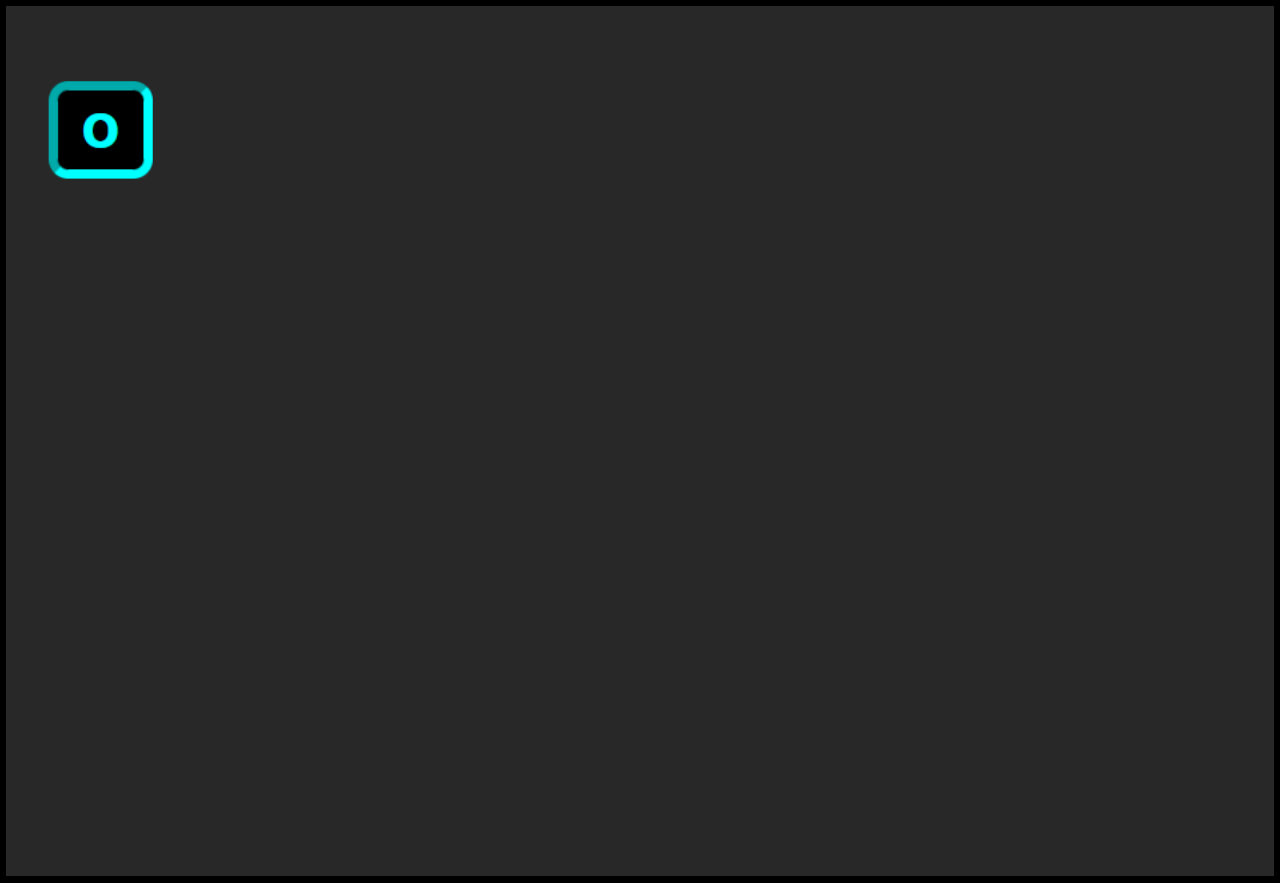
Christopher Andrew Topalian

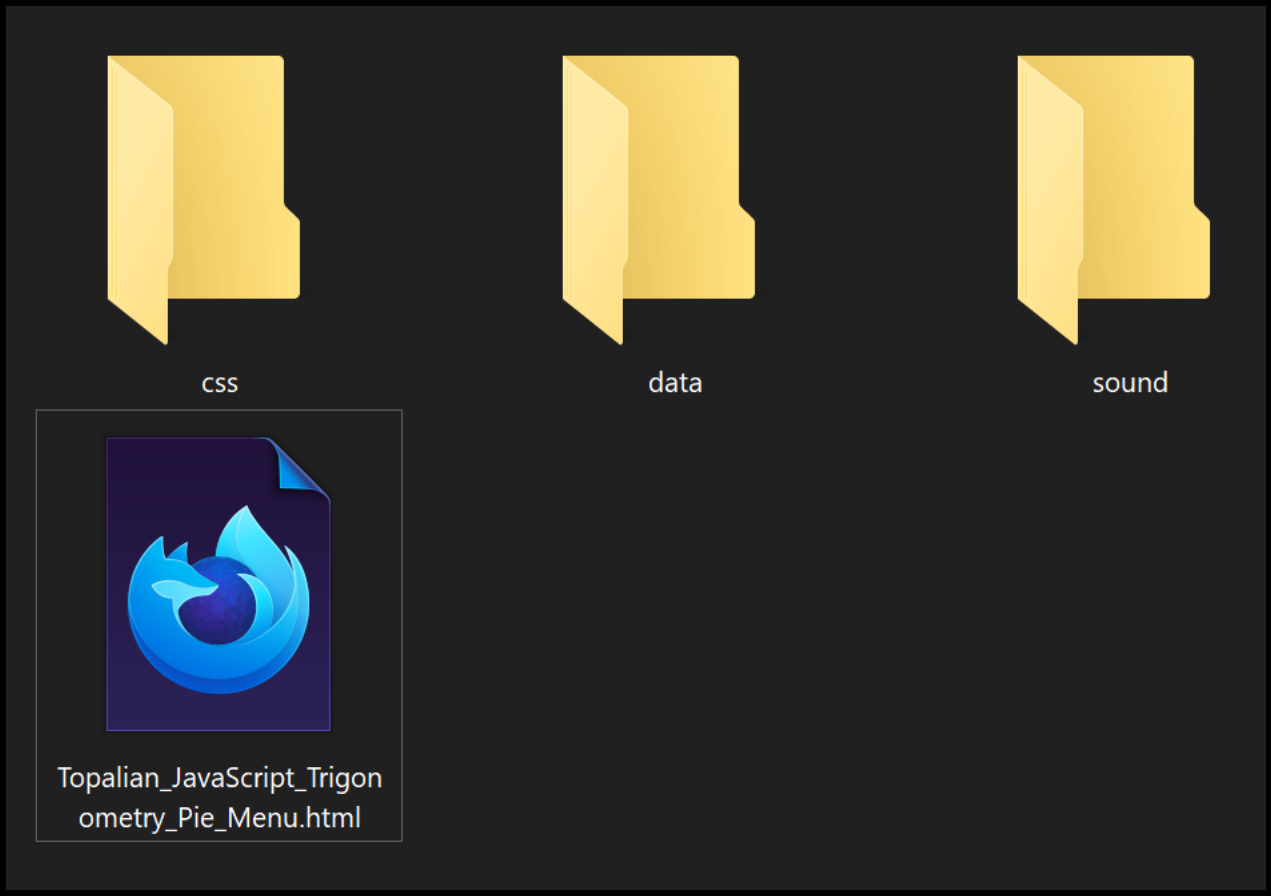
Dedicated to God the Father


O
Gate 1
Gate 2
Gate 3
Gate 4
Gate 5
Gate 6
Gate 7
Gate 8


O
Gate 1
Gate 2
Gate 3
Gate 4
Gate 5
Gate 6
Gate 7
Gate 8
Gate 7
Details of Gate 7


o
Gate 1
Gate 2
Gate 3
Gate 4
Gate 5
Gate 6
Gate 7
Gate 8
Gate 8
Details of Gate 8


o


css
data
sound
Topalian_JavaScript_Trigonometry_Pie_Menu.html

```
<!-- Dedicated to God the Father -->

<!-- All Rights Reserved Christopher Topalian
Copyright 2000-2024-->

<!-- https://github.com/ChristopherTopalian --
>

<!--
https://github.com/ChristopherAndrewTopalia
n -->

<!--
Topalian_JavaScript_Trigonometry_Pie_Men
u.html -->

<!-- Version 001 - (2024-01-26) -->
```

```
<html>
<head>
<title> Topalian JavaScript Trigonometry Pie
Menu </title>

<link rel = 'stylesheet' href =
'css/style001.css'>

<script src = 'data/logicGates.js'></script>

<script src = 'data/audio.js'></script>

<script>

function ge(whichId)
{
    let result =
document.getElementById(whichId);
```

```
    return result;
}

function ce(whichType)
{
    let result =
document.createElement(whichType);

    return result;
}

function ba(whichElement)
{
    let result =
document.body.append(whichElement);

    return result;
```

```
}

function audioLoad(whichArray)
{
    for (let x = 0; x < whichArray.length; x++)
    {
        let sound = ce('audio');
        sound.id = whichArray[x].id;
        sound.src = whichArray[x].src;
        ba(sound);
    }
}

function audioPlay(whichId, whichVolume)
{
    if (ge(whichId))
    {
        ge(whichId).volume = whichVolume;
```

```
        ge(whichId).play();
    }
}

function setStage()
{
    // buttonContainer
    let buttonContainer = ce('div');
    buttonContainer.id = 'buttonContainer';
    buttonContainer.className =
'buttonContainer';
    buttonContainer.style.position = 'absolute';
    buttonContainer.style.left = '100px';
    buttonContainer.style.top = '100px';
    ba(buttonContainer);

    //-//
```

```
    // buttonDivCircle
    let buttonDivCircle = ce('button');
    buttonDivCircle.id = 'buttonDivCircle';
    buttonDivCircle.className =
'buttonDivCircle';
    buttonDivCircle.innerHTML = 'O';

    //-//

    // onclick
    buttonDivCircle.onclick = function()
    {
        audioPlay('sfx_blip_001', 1.0);

        if (ge('circleContainer'))
        {
            ge('circleContainer').remove();
        }
```

```
        else
        {
            createCircleOfDivs(logicGates);
        }
    };
    buttonContainer.append(buttonDivCircle);
}

function createCircleOfDivs(whichArray)
{
    // circleContainer
    let circleContainer = ce('div');
    circleContainer.id = 'circleContainer';
    ba(circleContainer);

    //-//

    // infoDiv
```

```
    let infoDiv = ce('div');
    infoDiv.id = 'infoDiv';
    infoDiv.className = 'infoDiv';
    infoDiv.style.position = 'absolute';
    infoDiv.style.left =
ge('buttonContainer').getBoundingClientRect(
).x + 425;
    infoDiv.style.top =
ge('buttonContainer').getBoundingClientRect(
).y + 70;
    circleContainer.append(infoDiv);

    let radius = 100;

    let centerX =
ge('buttonContainer').getBoundingClientRect(
).x + 200;
```

```
    let centerY =
ge('buttonContainer').getBoundingClientRect(
).y + 100;

    //-//

    for (let i = 0; i < whichArray.length; i++)
    {
        let angle = (Math.PI * 2) /
whichArray.length * i - Math.PI / 2;

        let x = centerX + radius *
Math.cos(angle);

        let y = centerY + radius * Math.sin(angle);

        //-//
```

```
        // square
        let square = ce('div');
        square.innerHTML = whichArray[i].name;
        square.id = 'square' + i;
        square.className = 'square';
        square.style.left = x + 'px';
        square.style.top = y + 'px';

        // onclick
        square.onclick = function()
        {
            audioPlay('sfx_blip_001', 1.0);

            ge('infoDiv').innerHTML =
whichArray[i].name;

            ge('square' + i).style.color = 'rgb(255,
0, 255)';
```

```
        if (whichArray[i].name == 'Gate 1')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 2')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 3')
        {
```

```
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 4')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 5')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }
```

```
        if (whichArray[i].name == 'Gate 6')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 7')
        {
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }

        if (whichArray[i].name == 'Gate 8')
        {
```

```
            ge('infoDiv').innerHTML =
logicGates[i].name + '<br>' +
logicGates[i].details;
        }
    };

    // onmouseover
    square.onmouseover = function()
    {
        audioPlay('sfx_warp_001', 1.0);

        square.style.borderColor = 'rgb(0, 255,
255)';
        square.style.color = 'rgb(0, 255, 255)';
    };

    // onmouseout
    square.onmouseout = function()
```

```
        {
            square.style.borderColor = 'rgb(255,
255, 255)';
            square.style.color = 'rgb(255, 255,
255)';
        };

        circleContainer.append(square);
    }
}

function whenLoaded()
{
    audioLoad(audio);
    setStage();
}

</script>
```

```
</head>

<body onload = 'whenLoaded();'>

</body>

</html>
```

```
// Dedicated to God the Father

// All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024

// https://github.com/ChristopherTopalian

//
https://github.com/ChristopherAndrewTopalia
n

// audio.js

let audio =
[
    {
        name: 'sfx_blip_001',
        id: 'sfx_blip_001',
```

```
        src: 'sound/sfx_blip_001.mp4'
    },

    {
        name: 'sfx_warp_001',
        id: 'sfx_warp_001',
        src: 'sound/sfx_warp_001.mp3'
    }
];
```

```
// Dedicated to God the Father

// All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024

// https://github.com/ChristopherTopalian

//
https://github.com/ChristopherAndrewTopalia
n

// logicGates.js

let logicGates =
[
    {
        name: 'Gate 1',
        details: 'Details of Gate 1'
```

```
},

{
    name: 'Gate 2',
    details: 'Details  of Gate 2'
},

{
    name: 'Gate 3',
    details: 'Details  of Gate 3'
},

{
    name: 'Gate 4',
    details: 'Details  of Gate 4'
},

{
```

```
    name: 'Gate 5',
    details: 'Details  of Gate 5'
},

{
    name: 'Gate 6',
    details: 'Details  of Gate 6'
},

{
    name: 'Gate 7',
    details: 'Details  of Gate 7'
},

{
    name: 'Gate 8',
    details: 'Details  of Gate 8'
}
```

```
];
```

```
/* Dedicated to God the Father */

/* All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024 */

/* https://github.com/ChristopherTopalian */

/*
https://github.com/ChristopherAndrewTopalia
n */

/* style001.css */

body
{
    background-color: rgb(40, 40, 40);
    color: rgb(255, 255, 255);
    overflow: none;
```

```
}

.buttonContainer
{
    display: flex;
    flex-direction: row;
}

.square
{
    position: absolute;
    width: 45px;
    height: 45px;
    padding-left: 3px;
    padding-right: 3px;
    padding-top: 5px;
    border: 1px solid rgb(255, 255, 255);
    border-radius: 5px;
```

```
    background-color: rgb(0, 0, 0);
    text-align: center;
    line-height: 20px;
}

.square:hover
{
    border-style: solid;
    border-width: 2px;
    cursor: pointer;
}

.square:active
{
    cursor: grabbing;
}

.buttonDivCircle
```

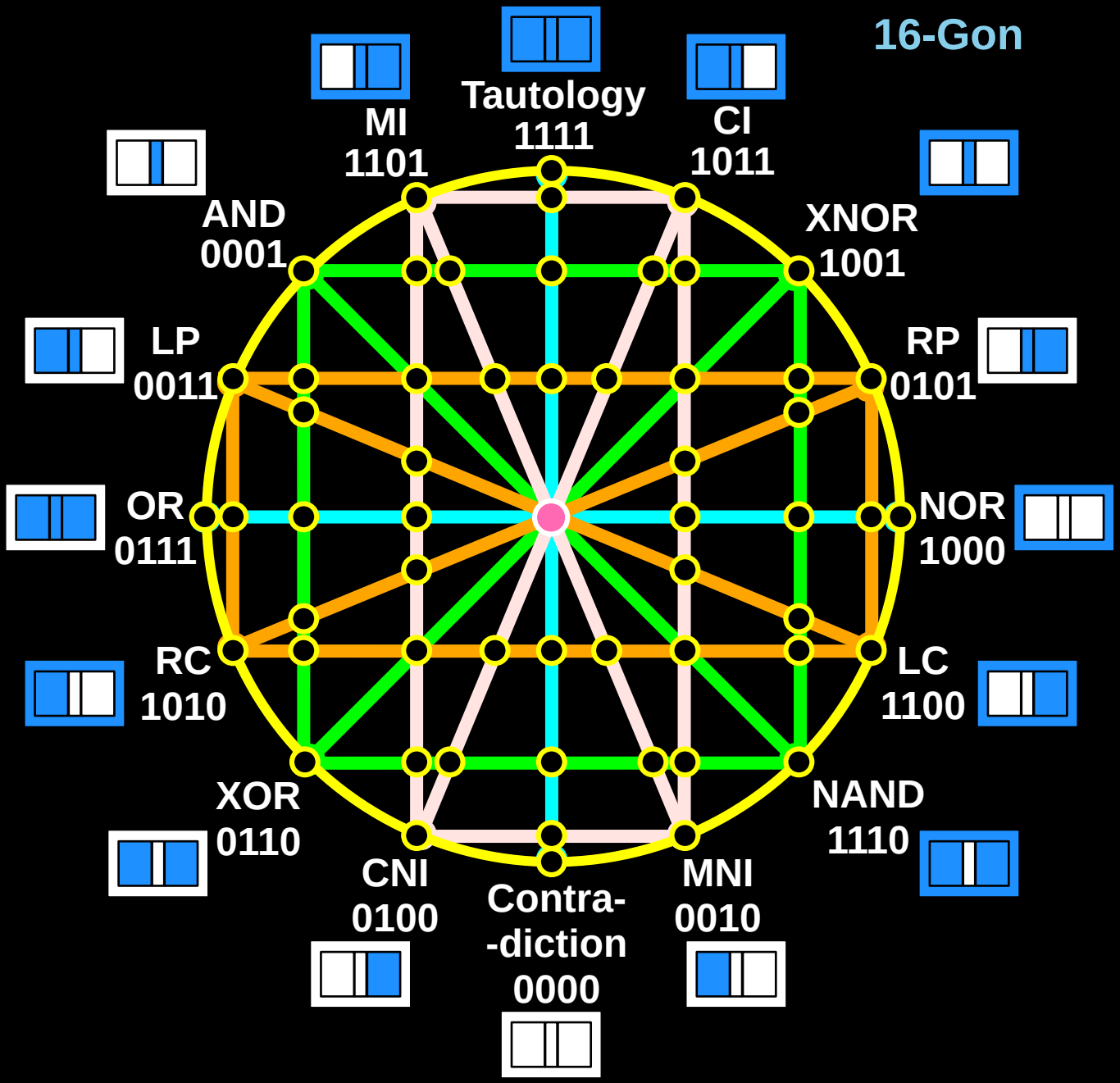
```
{
    padding-left: 10px;
    padding-right: 10px;
    padding-top: 4.2px;
    padding-bottom: 4px;
    border-radius: 8px;
    background-color: rgb(0, 0, 0);
    font-size: 20px;
    font-weight: bold;
    color: rgb(255, 255, 255);
}

.buttonDivCircle:hover
{
    border-color: rgb(0, 255, 255);
    cursor: pointer;
}
```

```
.buttonDivCircle:active
{
    border-color: rgb(0, 255, 255);
    border-width: 4px;
    color: rgb(0, 255, 255);
    cursor: grabbing;
}
```

Christopher Topalian

True Artificial Intelligence System



For More Tutorials:

CollegeOfScripting.weebly.com

CollegeOfScripting.wordpress.com

Youtube.com/ScriptingCollege

Twitter.com/CollegeOfScript

GitHub.com/ChristopherTopalian

GitHub.com/ChristopherAndrewTopalian

Sites.google.com/view/CollegeOfScripting

Dedicated to God the Father

This book is created by the
College of Scripting Music & Science.
Always remember, that each time you write a script with a pencil and paper, it becomes imprinted so deeply in memory that the material and methods are learned extremely well.

When you Type the scripts, the same is true. The more you type and write out the scripts by keyboard or pencil and paper, the more you will learn programming!

Write and Type every example that you find.
Keep all of your scripts organized.
Every script that you create increases your programming abilities.
SEEING CODE, is one thing,
but WRITING CODE is another.
Write it, Type it, Speak It, See It, Dream It.

CollegeOfScripting.weebly.com