

```

//cell.life v.1.82 based on the rules "game of life" by john horton conway
//cell.life v.1.82 programmed by printschler.j.m (action script 2.0)
//
// create the world
_root.onLoad = function() {
    _global.zae = 1;
    _global.tim = 1;
    _root.lif.text = "loading cell.life...";
    _global.cell_size = 10;
    _global.anz = 51;//World with 50*50 cells
    for (x=1; x<_global.anz; x++) {
        for (y=1; y<_global.anz; y++) {
            _root.createEmptyMovieClip("org_"+x+"_"+y,_root.getNextHighestDepth());
            _root["org_"+x+"_"+y]._visible = 0;
            rnd = random(50);
            if (rnd<5) {
                _root["org_"+x+"_"+y]._visible = 1;
            }
            _root["org_"+x+"_"+y]._x = (_global.cell_size*x)-_global.cell_size;
            _root["org_"+x+"_"+y]._y = (_global.cell_size*y)-_global.cell_size;
            _global.zelle.call(_root["org_"+x+"_"+y]);// function to visualize the cell
            _global.life_func.call(_root["org_"+x+"_"+y]);// lifefunction for each cell
            _global.borderfunc.call(_root["org_"+x+"_"+y]);
        }
    }
};
_root.onEnterFrame = function() {
    _global.tim = _global.tim+1;
    _root.lif.text = "23/3 system_duration: "+_global.tim;

};
////////////////////////////////////
// how would the cell look like
_global.zelle = function() {

    this.createEmptyMovieClip("cell",1);
    this.arra = new Array();
    this.arra[0] = x;
    this.arra[1] = y;
    with (this.cell) {
        lineStyle(0.25,0xfffff,100);
        moveTo(2.5,0);
        beginFill(0xfffff,0);
        lineTo(7.5,0);
        lineTo(10,2.5);
        lineTo(10,7.5);
        lineTo(7.5,10);
        lineTo(2.5,10);
        lineTo(0,7.5);
        lineTo(0,2.5);
        endFill();
        lineTo(2.5,0);
    }
    this.cell._yscale = 100;
    this.cell._xscale = 100;
};
////////////////////////////////////
_global.life_func = function() {

    this.onEnterFrame = function() {

        // PROGRAMMING THE MATRIX FOR EACH CELL
    }
};

```

```

if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1]-1)]._visible == false) {
    a = 0;
}
if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1]-1)]._visible == true) {
    a = 1;
}
if (_root["org_"+(this.arry[0])+"_"+(this.arry[1]-1)]._visible == false) {
    b = 0;
}
if (_root["org_"+(this.arry[0])+"_"+(this.arry[1]-1)]._visible == true) {
    b = 1;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1]-1)]._visible == false) {
    c = 0;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1]-1)]._visible == true) {
    c = 1;
}
}
////////////////////////////////////
if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1])]._visible == false) {
    d = 0;
}
if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1])]._visible == true) {
    d = 1;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1])]._visible == false) {
    e = 0;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1])]._visible == true) {
    e = 1;
}
}
////////////////////////////////////
if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1]+1)]._visible == false) {
    f = 0;
}
if (_root["org_"+(this.arry[0]-1)+"_"+(this.arry[1]+1)]._visible == true) {
    f = 1;
}
if (_root["org_"+(this.arry[0])+"_"+(this.arry[1]+1)]._visible == false) {
    g = 0;
}
if (_root["org_"+(this.arry[0])+"_"+(this.arry[1]+1)]._visible == true) {
    g = 1;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1]+1)]._visible == false) {
    h = 0;
}
if (_root["org_"+(this.arry[0]+1)+"_"+(this.arry[1]+1)]._visible == true) {
    h = 1;
}
}
////////////////////////////////////
this.moore = (a+b+c+d+e+f+g+h);
trace(this.moore);
////////////////////////////////////
// call the life function
_global.game_23_3.call(this);// based on conways original
//_global.game_1357_1357.call(this);
//_global.game_34_3.call(this);

```

```

};
_global.borderfunc = function() {

```

```

////////////////////////////////////
//border again
// left-top
if (this.arry[0]<=1) {
    this._visible = 0;
}
if (this.arry[1]<=1) {
    this._visible = 0;
}
// right-bottom
if (this.arry[0] == 50) {
    this._visible = 0;
}
if (this.arry[1] == 50) {
    this._visible = 0;
}
////////////////////////////////////
};
//LIFE FUNCTIONS
////////////////////////////////////
// conways rules of life a 23/3 system
_global.game_23_3 = function() {
    //////////////////////////////////////
    if (this._visible == 1) { //life
        if (this.moore<2) {
            this._visible = 0;
        }
        else if (this.moore>3) {
            this._visible = 0;
        }
    }
    //////////////////////////////////////
    else if (this._visible == 0) { // dead

        if (this.moore == 3) {
            this._visible = 1;
        }
    }
    _global.borderfunc.call(this);
};
////////////////////////////////////
_global.game_34_3 = function() {
    if (this._visible == 1) { //life

        //////////////////////////////////////

        if (this.moore<3) {
            this._visible = 0;
        }
        if (this.moore>4) {
            this._visible = 0;
        }
        if (this.moore == 3) {
            this._visible = 1;
        }
        if (this.moore == 4) {
            this._visible = 1;
        }
    }
    if (this._visible == 0) { // dead
        if (this.moore == 3) {
            this._visible = 1;
        }
    }
}

```

```

    }
    if (this.moore<3) {
        this._visible = 0;
    }
    if (this.moore>3) {
        this._visible = 0;
    }
    //////////////////////////////////////
}
_global.borderfunc.call(this);
};
_global.game_1357_1357 = function() {
    if (this._visible == 1) { //life

        //////////////////////////////////////

        if (this.moore == 1) {
            this._visible = 1;
        }
        if (this.moore == 2) {
            this._visible = 0;
        }
        if (this.moore == 3) {
            this._visible = 1;
        }
        if (this.moore == 4) {
            this._visible = 0;
        }
        if (this.moore == 5) {
            this._visible = 1;
        }
        if (this.moore == 6) {
            this._visible = 0;
        }
        if (this.moore == 7) {
            this._visible = 1;
        }
        if (this.moore == 8) {
            this._visible = 0;
        }
    }
    if (this._visible == 0) { // dead
        if (this.moore == 1) {
            this._visible = 1;
        }
        if (this.moore == 2) {
            this._visible = 0;
        }
        if (this.moore == 3) {
            this._visible = 1;
        }
        if (this.moore == 4) {
            this._visible = 0;
        }
        if (this.moore == 5) {
            this._visible = 1;
        }
        if (this.moore == 6) {
            this._visible = 0;
        }
        if (this.moore == 7) {
            this._visible = 1;
        }
    }
}

```

```
        if (this.moore == 8) {
            this._visible = 0;
        }
        //////////////////////////////////////
    }
    _global.borderfunc.call(this);
};
```