

PACMAN

Was wir bisher haben:

Main:

```
float rectSize = 40;
Pacman pac;

void settings() {
    size(400, 400);
}

void setup() {
    background(0);

    // erstelle pacman
    pac = new Pacman(width/2, height/2, rectSize);
}

void draw() {
    // zeichne hintergrund
    background(0);
    // zeichne pacman
    pac.paint();
}

void keyPressed() {
    if (keyCode == UP) {
        pac.moveUp();
    } else if (keyCode == DOWN) {
        pac.moveDown();
    } else if (keyCode == LEFT) {
        pac.moveLeft();
    } else if (keyCode == RIGHT) {
        pac.moveRight();
    }
}

class Pacman{
    color yellow = color(255, 255, 0);
    int speed = 8;
    float diam;
    float x;
    float y;

    //Konstruktor
    Pacman(float x, float y, float size){
        this.x = x;
        this.y = y;
        this.diam = size;
    }

    void paint(){
        noStroke();
        fill(yellow);
        ellipse(x, y, diam, diam);
    }
}
```

```

void moveLeft(){
    x = x + -1*speed;
}

void moveRight(){
    x = x + speed;
}

void moveUp(){
    y = y + -1*speed;
}

void moveDown(){
    y = y + speed;
}

float getX(){
    return x;
}

float getY(){
    return y;
}

float getSize(){
    return diam;
}
}

```

Wie wir unser Spielfeld aufbauen wollen:

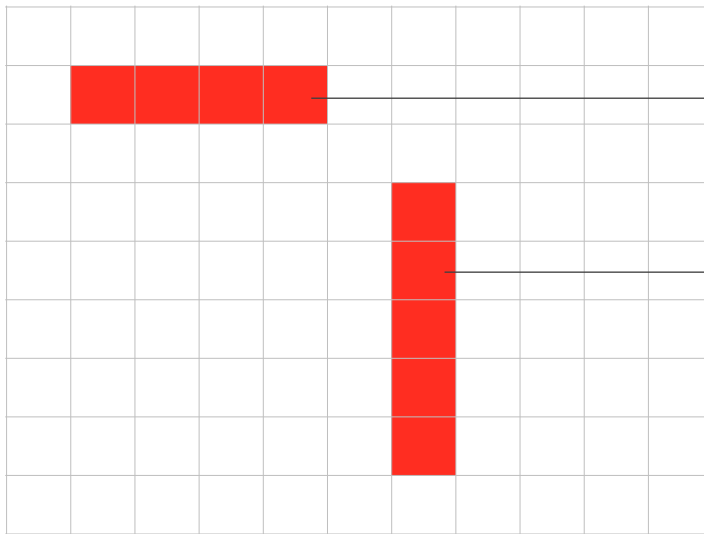
Um das Labyrinth an Hindernissen und die Nuggets die es einzusammeln gilt zu zeichnen, unterteilen wir unser Spielfeld im Kopf in ein Grid. Jedes Gridelement hat dabei die Grösse rectSize x rectSize, welches unsere Grundeinheit darstellen wird:

Spielfeld:



rectSize x rectSize

Nun können wir Hindernisse erstellen, welche eine Breite von `rectSize` haben und eine Länge von `x*rectSize`. Sie können vertical oder horizontal angeordnet sein.



Hindernis: horizontal, 4 Einheiten lang
= `4*rectSize`.

Hindernis: vertical, 5 Einheiten lang.

AUFGABE 1 - Hindernisse

Wir wollen Hindernisse / Labyrinthwände erstellen durch die Pacman (und später auch die Gegner) nicht durchfahren können.

Ergänze dazu die folgende Klasse für die Hindernisse:

```
class Obstacle{
    // Klassenattribute für Position, Länge und Breite.
    // Und isVertical zum definieren ob das Rechteck liegend oder stehend
    // gezeichnet wird.

    // Konstruktor
    Obstacle(float x, float y, boolean isVertical, float obLength, float
rectSize){
        // Initialisieren die Position und ob das Hindernis vertical oder
        // horizontal liegt.
        // Initialisiere die Hindernis Breite und Länge. Diese sind abhängig
        // davon wie das Hindernis liegt.
        // Die längere Seite soll in Einheiten, nicht der effektiven Länge,
        // übergeben werden und muss hier nun wieder in
        // eine Länge umgerechnet werden also: obLength * rectSize
    }

    // Methode zum zeichnen des Rechtecks
    void paint(){
    }

    // Methode die true zurück gibt, wenn die x UND y Parameter das Rechteck
    // berühren
    // oder innerhalb des Rechtecks liegen.
    boolean isTouched(float x, float y){
    }
}
```

Erstelle nun einen Array von Hindernissen in Main, lasse sie zeichnen und teste ob dein Pacman nicht drüber fahren kann.

Sobald das klappt, kannst du eine Klasse `Playfield` erstellen, welche ab jetzt die Hindernisse verwalten soll. Ergänze dazu diese Klasse:

```
class PlayField {
    // Attribut für die Hindernisse (Array von Hindernissen)
    // und die rectSize

    PlayField(float rectSize) {
        this.rectSize = rectSize;
        initObstacles();
    }

    void initObstacles() {
        // erstelle hier deine Hindernisse
    }

    void paint(){
        // zeichne die Hindernisse
    }

    boolean isObstacleTouched(float x, float y){
        // prüfe ob eines der Hindernisse berührt wurde
    }
}
```

Zusatz: Erstelle deine Hindernisse so, dass dein Spielfeld so aussieht:
Und setze deinen Pacman in die linke obere Ecke des Spielfelds.

