

Lekcja 7 – Grawitacja – Zdarzenia - HTML

plotkarka.eu (kodujący stwór, lekcja 55)

Jak zasymulować grawitację? Animacja nie jest problemem, a grawitacja w przypadku Ziemi i wszystkich znajdujących się na niej obiektów polega na tym, że, gdy je podrzucimy, to zaraz spadają, z coraz większą prędkością – z przyspieszeniem.

Wszystkie równania, które odpowiadają za poruszanie się obiektów można wyprowadzić z najprostszych fizycznych równań na prędkość i przyspieszenie. Po niewielkich przekształceniach można je zapisać w następujący sposób:

A oto cały program umożliwiający symulację wszystkich podstawowych rzutów oraz drgań:

```
//równania ruchu
x = x + vx * dt;
y = y + vy * dt;
//przyspieszenie
vy = vy + g * dt;
//opór powietrza
vx = vx * opor;
vy = vy * opor;
//sprężyste odbicia
vy = vy * spr;
```

Promień piłki i korekta odbicia

Możemy zauważyć, że z każdym odbiciem piłka leci coraz wyżej – prędkość v_x zwiększa się dodatkowo po każdym odbiciu. Warunek odbicia poddamy korekcie i ujmemy dodatkowy wzrost prędkości.

Poprawimy również samo odbicie – piłka nie będzie „wchodziła” w ziemię, gdy zmniejszymy brzeg o promień.

```
if (y>h-r-vy*dt){vy=- (vy-g*dt)*spr}
```

Przycisk OD NOWA

Wcisnąć za każdym razem F5, aby zobaczyć piłkę od nowa, to nie jest dobre rozwiązanie. Prawdziwy program reaguje na myszkę i klawiaturę – w JS odpowiadają za te funkcje tzw. **zdarzenia**.

Najpierw zdefiniujemy w HTML **przycisk** i przypiszemy do zdarzenia **onclick** funkcję:

```
<input
  type="button"
  value="START"
  onclick= OdNowa()
>
```

Funkcja **OdNowa** zwiera dane początkowe i uruchomienie animacji:

```
function OdNowa(){
  x=20;  y=20;
  vx=10; vy=0;
  czas=setTimeout(grawitacja,skok);
}
```

Rysowanie – wymazywanie

Pole HTML **checkbox** posłuży nam do ustawiania wymazywania ekranu podczas animacji.

Wymazywanie ekranu będzie

teraz zależało od stanu

pola checkbox o nazwie **rys0**

```
czyść ekran
<input
  id="rys0"
  type="checkbox"
>
```

```
if (rys0.checked) {
  c.clearRect(0, 0, w, h);
}
```

Suwak oporu powietrza

Za pomocą suwaka **RopS** określamy opór powietrza, opór wpisujemy po każdej

zmianie – zdarzenie **onchange** do pola tekstowego **RopT**.

Funkcja **OporPow** oblicza na nowo parametr i wpisuje go do pola tekstowego.

```
opór powietrza
<input type="range" id="RopS" min=990 max=1000 onchange=OporPow()>
<input type="text" id="RopT" size=2>
```

```
function OporPow(){
  opor=RopS.value/1000;
  RopT.value=opor;
}
```

```
// kolorowa piłka
function pilka(x,y,r,kolor) {
  c.beginPath();
  c.strokeStyle=kolor;
  c.fillStyle=kolor;
  c.arc(x,y,r,0,2*Math.PI);
  c.stroke();
  c.fill();
};
// szerokość i wysokość canvas
var w = c.canvas.width;
var h = c.canvas.height;
// parametry animacji
var skok = 10;
var czas;
// parametry ruchu
var x=20;
var y=20;
var vx=10;
var vy=0;
var r=5;
var kolor="red";
var g=9.81; //grawitacja
var dt=0.1; //skok czasu
var opor=0.999; //opór powietrza
var spr=0.94; //sprężystość
// rekurencyjna animacja
function grawitacja() {
  c.clearRect(0, 0, w, h);
  c.strokeStyle="black";
  c.strokeRect(0, 0, w, h);
  pilka(x,y,r,kolor);
  x=x+vx*dt;
  y=y+vy*dt;
  vy=vy+g*dt;
  vx=vx*opor;
  vy=vy*opor;
  if (y>h){vy=-vy*spr}
  clearTimeout(czas);
  czas = setTimeout(grawitacja, skok);
}
grawitacja();
```