# Лабораторная работа № 4. Движение тела, брошенного под углом к горизонту

## Задача

Изобразить местонахождение тела, брошенного с земли с начальной скорость под углом к горизонту , через равные интервалы времени ( – число равных временных интервалов).

## Математическая модель

Координаты произвольной точки траектории движения определяются уравнениями:

Для определения времени полета прировняем :

Тогда длительность временного интервала (шаг по ):

## Код программы

WD = 500  
HG = 250  
ZR\_X = 10  
ZR\_Y = 240  
g = 9.8  
  
  
def draw\_axis():  
 cnvs.create\_text(WD - 20, ZR\_Y - 20, text='X')  
 cnvs.create\_text(ZR\_X + 20, 20, text='Y')  
 x\_vals = [i/10 for i in range(21)]  
 y\_vals = [i/10 for i in range(11)]  
 x\_cords = [ZR\_X+x\_vals[i]\*240 for i in range(len(x\_vals))]  
 y\_cords = [ZR\_Y-y\_vals[i]\*230 for i in range(len(y\_vals))]  
 for i in range(len(x\_cords)):  
 cnvs.create\_oval(x\_cords[i], ZR\_Y, x\_cords[i] + 2, ZR\_Y + 2, fill='black')  
 for j in range(len(y\_cords)):  
 cnvs.create\_oval(ZR\_X, y\_cords[j], ZR\_X + 2, y\_cords[j] + 2, fill='black')  
  
  
def x\_func(t):  
 v = int(v0.get())  
 ang = int(angle.get()) \* math.pi/180  
 return v \* t \* math.cos(ang)  
  
  
def y\_func(t):  
 v = int(v0.get())  
 ang = int(angle.get()) \* math.pi/180  
 return v \* t \* math.sin(ang) - g \* t \* t / 2  
  
  
def draw():  
 cnvs.delete("all")  
 cnvs.create\_line(0, 240, 490, 240, arrow=LAST)  
 cnvs.create\_line(10, 250, 10, 10, arrow=LAST)  
 draw\_axis()  
 tm = float(time.get())  
 i = 0  
 xc = 0  
 yc = 0  
 while i <= tm:  
 xc = round(ZR\_X + x\_func(i)\*100)  
 yc = round(ZR\_Y - y\_func(i)\*100)  
 if yc <= 240:  
 cnvs.create\_oval(xc, yc, xc + 1, yc + 1)  
 i += tm/1000  
 print(xc)  
 print(yc)

## Результаты



