

**Московский государственный технический
университет им. Н.Э. Баумана**

**Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Парадигмы и конструкции языков программирования»

Отчет по домашнему заданию

Выполнил:
студент группы ИУ5-34Б:

Шимко Даниил
Подпись и дата:

Проверил:
преподаватель каф.
ИУ5
Гапанюк Ю.
Подпись и дата:

Москва, 2023 г.

Задание:

Написать любую игру(в моем случае написана змейка на питоне)

Код:

```
import tkinter
```

```
from tkinter import *
```

```
from random import randint
```

```
#  
=====
```

```
# ===== объявление собственных  
=====
```

```
# ===== функций  
=====
```

```
#  
=====
```

```
# Место для функций, которые будут реагировать на нажатия клавиш
```

```
# import button as button
```

```
def left(a):
```

```
    global dx, dy
```

```
if not dx:
```

```
    dx = -side
```

```
    dy = 0
```

```
def right(a):
```

```
    global dx, dy
```

```
    if not dx:
```

```
        dx = side
```

```
        dy = 0
```

```
def up(a):
```

```
    global dx, dy
```

```
    if not dy:
```

```
        dy = -side
```

```
        dx = 0
```

```
def down(a):
```

```
    global dx, dy
```

```
    if not dy:
```

```
        dy = side
```

```
dx = 0
```

```
def start_game(t):
```

```
    global space_debug
```

```
    if space_debug:
```

```
        redraw()
```

```
        space_debug = not space_debug
```

```
# Место для функции redraw(), которая будет обновлять экран
```

```
def redraw():
```

```
    global x, y, b, a, bits
```

```
    if flag:
```

```
        x += dx
```

```
        x %= width
```

```
        y += dy
```

```
        y %= height
```

```
    if [x, y] in bits:
```

```
        canv.delete(ALL)
```

```
        canv.create_text(375, 375, text="GAME OVER", justify=CENTER,  
font="Courier 50")
```

```
        canv.create_text(375, 475, text="SCORE: " + str(a), justify=CENTER,  
font="Courier 35")
```

```
    return
```

```
bits.append([x, y])
```

```
if x == xap and y == yap:
```

```
    apple()
```

```
    a += 1
```

```
    if b > 20:
```

```
        b -= 10
```

```
else:
```

```
    del bits[0]
```

```
canv.delete(ALL)
```

```
for xbit, ybit in bits:
```

```
    canv.create_oval(xbit, ybit, xbit + side, ybit + side, fill="IndianRed1")
```

```
    canv.create_text(60, 10, text="Съедено яблочко: " + str(a),  
justify=CENTER, font="Times 10")
```

```
canv.create_oval(xap, yap, xap + side, yap + side, fill="green2")
```

```
main.after(b, redraw)
```

Ниже будет функция для создания случайных координат яблока

def apple():

 global xap, yap

 xap = randint(0, width - side) // side * side

 yap = randint(0, height - side) // side * side

 while [xap, yap] in bits:

 apple()

Здесь напишем функцию для режима "пауза"

def pause():

 global flag

 flag = not flag

#

=====

===== игровая логика

=====

#

=====

Блок, отвечающий за создание окна и расстановки в нём виджетов

```
main = Tk()
```

```
Button = Button(text="Pause", command=pause)
```

```
width, height = 760, 760
```

```
canv = Canvas(width=width, height=height, bg="aquamarine2")
```

```
canv.create_text(350, 350, text="SNAKE", justify=CENTER, font="Courier  
40")
```

```
canv.create_text(350, 500, text="press SPACE to start", justify=CENTER,  
font="Courier 25")
```

```
canv.pack()
```

```
Button.pack()
```

```
# Блок, в котором мы создаём необходимые переменные и константы
```

```
side = 20
```

```
x, y = 20, 20
```

```
dx, dy = side, 0
```

```
start_pack = 3
```

```
bits = []
```

```
xap = yap = 0
```

```
b = 300
```

```
flag = True
```

```
a = 0
```

```
c = 1
```

```
t = 0
```

```
space_debug = True
```

```
# Место сотворения змейки в её первоизданном виде
```

```
for i in range(start_pack):
```

```
    bits.append([x + i * side, y])
```

```
x, y = bits[-1][0], bits[-1][1]
```

```
apple()
```

```
# Назначаем бинды кнопок
```

```
main.bind("<w>", up)
```

```
main.bind("<s>", down)
```

```
main.bind("<a>", left)
```

```
main.bind("<d>", right)
```

```
main.bind("<Up>", up)
```

```
main.bind("<Down>", down)
```

```
main.bind("<Left>", left)
```

```
main.bind("<Right>", right)
```

```
main.bind("<space>", start_game)
```

```
mainloop()
```