**Remastering and development of classic games based on Python**

**Class: Digital Media 2003**

**Name: Hao Chu**

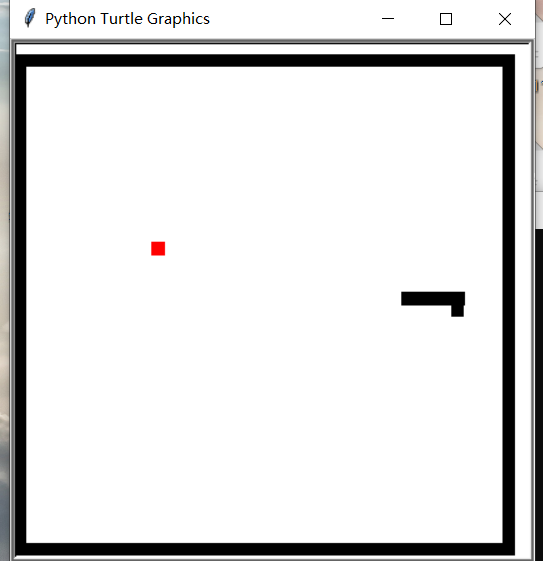
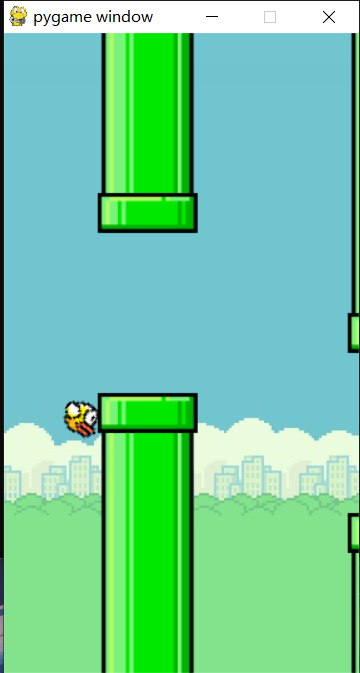
**Student ID: 1191200328**

1. **introduction**

**Game can refer to** [**both a**](https://zh.wikipedia.org/wiki/%E4%BA%BA) **kind of** [**entertainment**](https://zh.wikipedia.org/wiki/%E5%A8%B1%E4%B9%90) [**activity**](https://zh.wikipedia.org/wiki/%E6%B4%BB%E5%8B%95) **of people and the process of such activities. A game is a type** [**of organized play**](https://zh.wikipedia.org/wiki/%E7%8E%A9%E8%80%8D)**, generally** [**for the**](https://zh.wikipedia.org/wiki/%E5%A8%9B%E6%A8%82) **purpose of entertainment and sometimes** [**for educational**](https://zh.wikipedia.org/wiki/%E6%95%99%E8%82%B2) **purposes. Games are different from jobs that have monetary rewards, and they are not like art that presents aesthetic or conceptual elements. However, the demarcation between each other is not necessarily clear, as play and work as a professional athlete may be one, while** [**jigsaw puzzles**](https://zh.wikipedia.org/wiki/%E6%8B%BC%E5%9C%96) **have both game and art elements. [[1]](#endnote-1)**In the 21st century of technological development, video games are like a rising star, sweeping the world and infiltrating every aspect of our lives. While bringing huge economic benefits, it also greatly enriches the audio-visual experience of players. It is for this reason that video games are rated as the "ninth art" by the people. A triple-A work (high-cost, high-volume, high-quality stand-alone game), whether it is its exquisite graphics, modeling, or its extreme depiction of building details and infectious music, to the gripping plot development, all embody the charm of painting, sculpture, architecture, music, film, literature and other arts.

The post-90s and post-00s generations can be said to have witnessed the development of video games, from the game disc when they were children to the current next-generation console, from Contra to Call of Duty, from the Green Corps to Assassin's Creed. A lot of players have grown up with the game. At the moment when there are a variety of 3A masterpieces. We may have forgotten the Super Mario on the red and white plane, or the snake on an elderly plane. But as with any great cause, the early pioneers are called pioneers. Early achievements are known as milestones. But these classic games are no longer compatible with today's devices or have more modern versions. Many classic games are placed in the depths of memory just like our childhood. Classics shouldn't just go away, and the development of high-level programming languages and game engines has made it possible to remake classic games. In this semester, I have a basic understanding of the basic knowledge and characteristics of Python. The simplicity of the Python language and the rich and powerful libraries of Python make it easier to implement many complex functions. Python also has a powerful game library called Pygame, which makes Python game development much easier than other languages without the help of any engine.

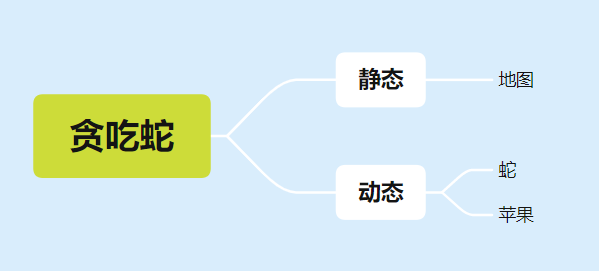
So I decided to use Python to recreate the classic game, Snake, and develop a classic Hopping Bird game.



1. **solution**

**2.1**

The first is a replica of the snake, which we can divide into three parts for a simple snake game, namely the snake, the apple, and the map. Looking at its properties, it can be roughly divided into dynamic snakes and constantly refreshing apples, as well as static maps. In contrast, static maps are much easier to implement, and the more complex ones are the movement of snakes and the refresh of apples.

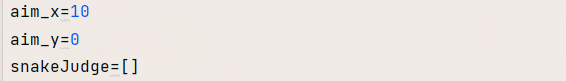


First of all, snakes and apples, we need to display them in the program window, then we need to call the turtle drawing in the Python library, we define a square function in the gamebase file, then the drawing operation is completed by this square function, we only need to import the gamebase in the real game file, in order to enhance the readability of the program.



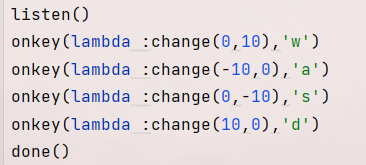
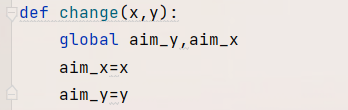
So in the display window, we use a queue to represent the snake and a dot to represent the apple.

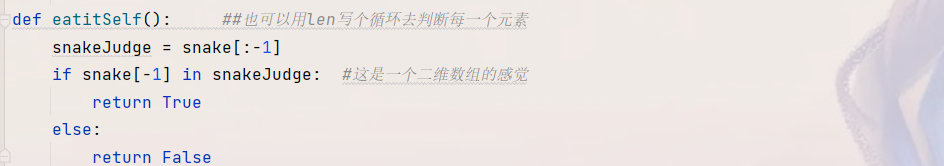
Apples are randomly generated, so we used the randrange function.



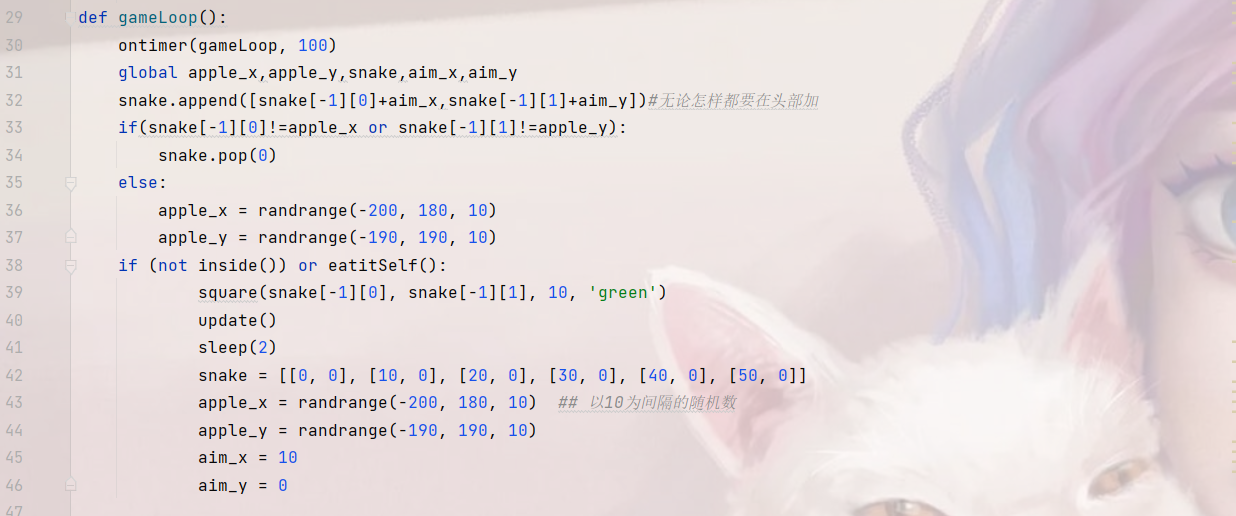
aim\_x and aim\_y are the two variables used to manipulate the direction of the snake, followed by a queue of snakeJudges to determine if the snake has bitten itself. Here's how to do it:

There is also the movement of the snake, which uses a listen() function to listen to the keyboard commands, and then changes the aim\_x or aim\_y to change the direction of the snake.

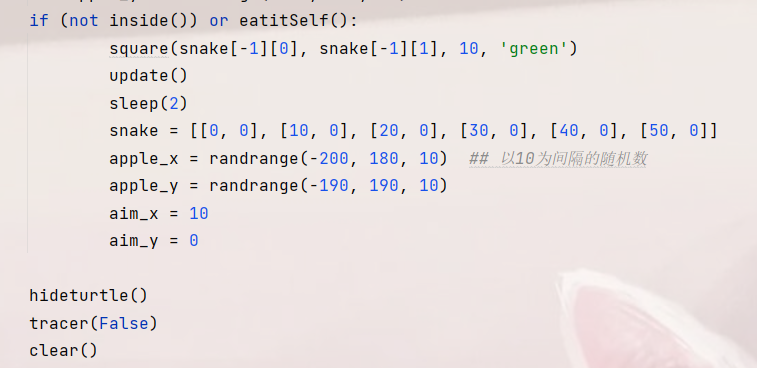




There is also an important logic for the movement of the snake, by constantly deleting the queue element and increasing the queue element to achieve the movement of the snake, that is, if the snake wants to move forward, I must add an element to the tail of the queue (snake head) every time, and delete an element at the head of the queue, the element is the pixel point, so as to realize the progress of the snake, and at the same time to make the judgment of eating the apple, that is, the coordinates of the snake head (team tail) overlap with the coordinates of the apple, so that the snake tail is not deleted to achieve the growth of the snake. The implementation code is as follows:



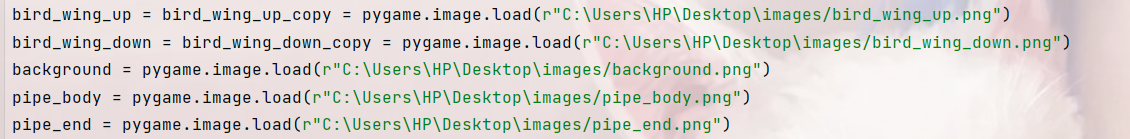
The last is the establishment of the map, in fact, the logic of the collision is very easy to implement, once the snake head reaches the regional boundary or the snake bites itself then the snake head turns green, the game is over, in order to be able to display the map boundary on the program interface, I drew a large (black) and a small (white) two squares, so as to hand over a square border, as the map boundary.



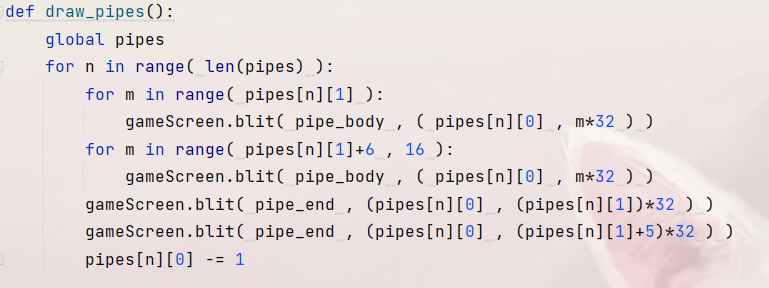


**2.2**

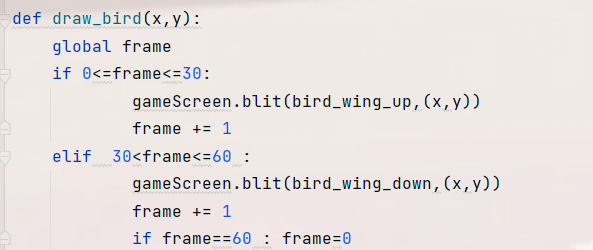
When developing Hopping Bird, I imported Pygame's library, which allowed the game to implement some more complex features, such as importing images, setting frame rates, etc., which greatly improved the integrity of the game.



Draw the pipeline



Draw birds



Determine if the bird has touched the ground and touched the pipe



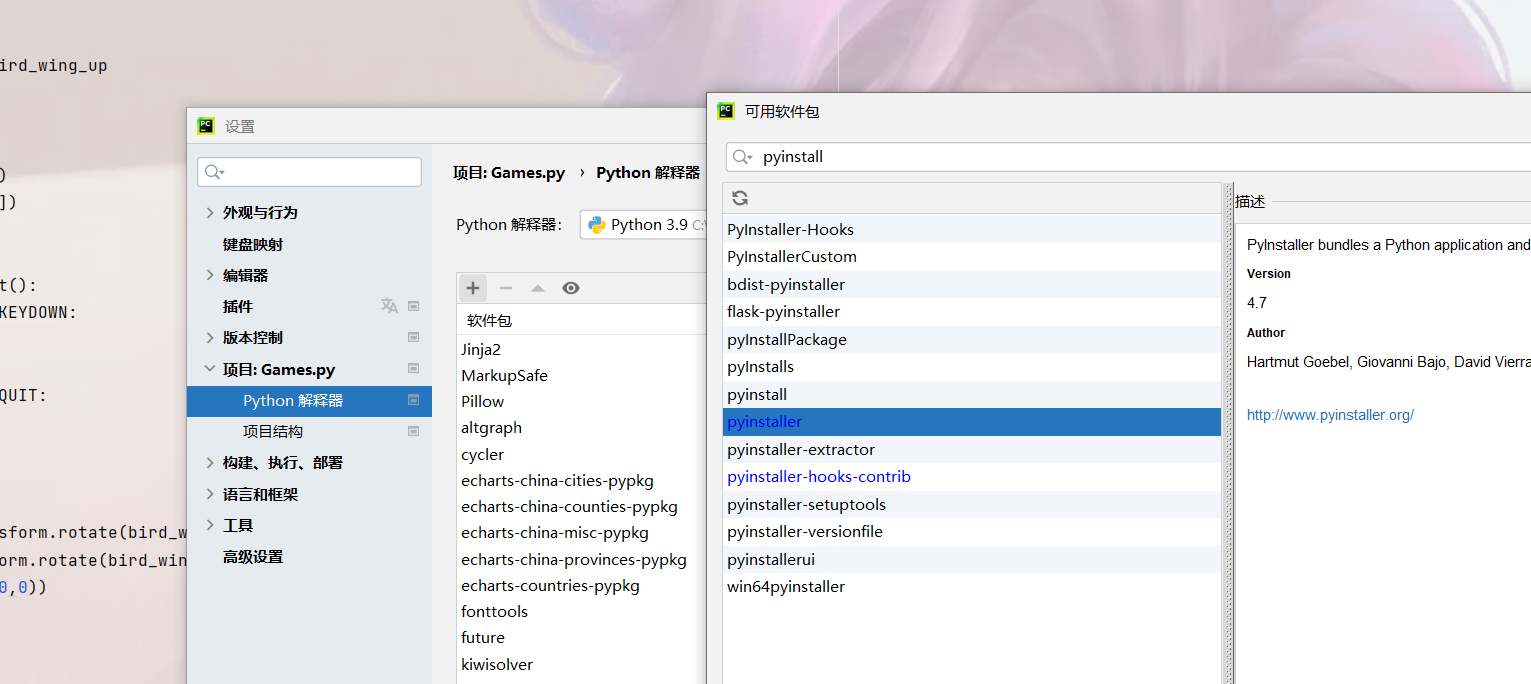
At the same time, an interesting feature is implemented, which is to use gravity to simulate the fall of a bird



At the same time, the bird flies up and down in different states (by rotation)



The last is the packaging of the game, the compiler I use is Pycharm, which is also very rich and powerful, click on the settings in the beginning, and add Pyinstaller to the Python interpreter.



Finally, enter the following instructions in the terminal to complete the packaging



1. **conclusion**

Through a series of tests and debugging, as well as the play of some players, it can be determined that the game can run normally, the reproduced game functions are more complete, the game runs more smoothly, the snake runs more smoothly when only calling the turtle drawing, and the jumping bird runs more smoothly when calling Pygame, and the amount of code is greatly reduced, and the logic between the various modules becomes clearer.

In the process of completing these two games, I really felt the simplicity and logic of the Python language, and at the same time, with the blessing of powerful libraries, it made the development of mini games much easier. However, after consulting the data, it is found that Python does not seem to be suitable for the development of large-scale games, there are gains and losses, Python in the acquisition of its language conciseness but also caused some syntax defects (omitting the declaration of variables), which leads to our complex program logic is not easy to understand the code, in many large-scale game development, due to the complexity of its functions, the amount of code required even reaches the level of millions, at this time the efficiency of using Python is far less than C++, C#等语言了. [[[2]](#endnote-2)]

But in general, the advantages of Python in the development of small-scale programs are still huge, and I can indeed feel many advantages of Python through the development of this game, I believe this is a starting point on my Python learning path.

1. **prospect**

This remake of the classic game makes me feel like I am back in my childhood. Compared with the past, the video game industry has undergone earth-shaking changes, whether it is the economic value it creates, or the cultural connotation and social identity it brings, it has made a non-negligible contribution to the development of human society, but every industry has shortcomings, and the same is true for the video game industry, but these shortcomings need to be faced and changed together. At the end of this semester, I realized that Python is a powerful language that has great prospects for development in artificial intelligence, pattern recognition, and big data analysis, and I believe that with the development of technology in the future, the metaverse will be popularized in our lives. In the future, the game will be more immersive, and I believe that with the power of Python, one day we will have a more realistic conversation with a realistic NPC in the game. From big data to pattern recognition, I believe Python will one day break down the barrier between the virtual and the real.

1. **References**

[1] Wikipedia's definition of a game https://zh.wikipedia.org/wiki/%E6%B8%B8%E6%88%8F

[2] Zhihu "Can Python Develop Games" https://zhuanlan.zhihu.com/p/152391793

1. [↑](#endnote-ref-1)
2. [↑](#endnote-ref-2)