

## Lab 3

Due 12:00, Wednesday, July 28, 2021

### No Late Submission

注意事項：

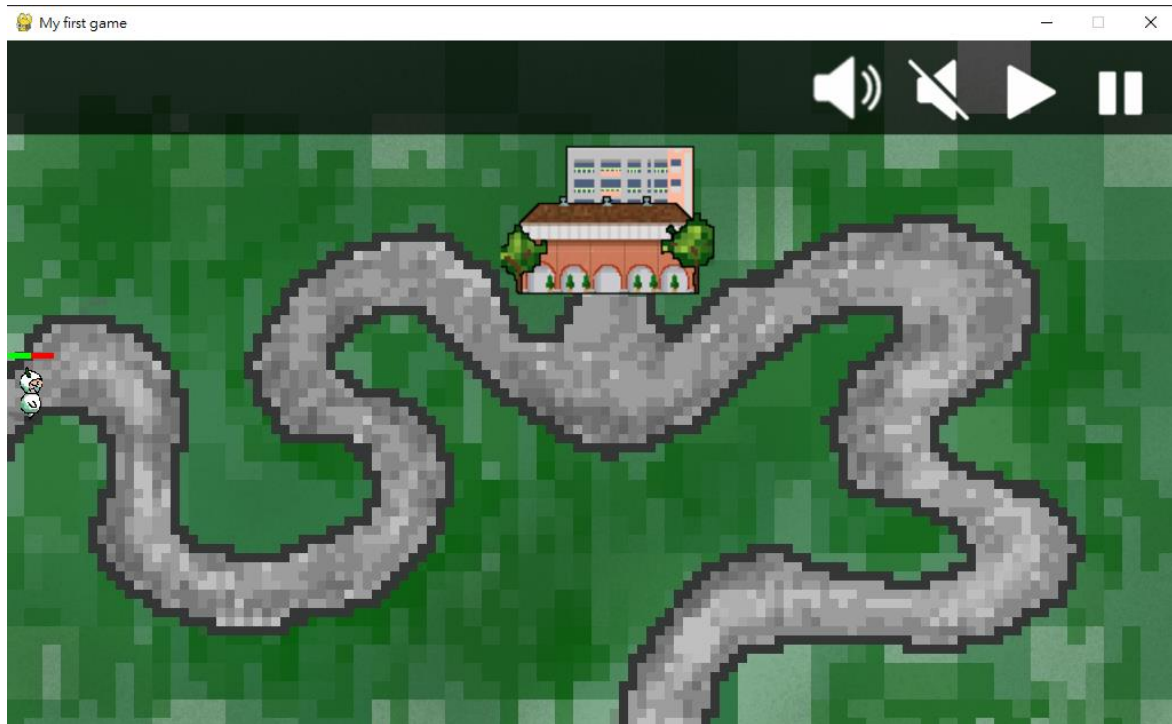
1. Lab 的時間為授課結束後至下午一點。
2. Lab 的分數分配：出席 20%，三題練習 80%，Bonus 50%。
3. 請儘量於 Lab 時段完成練習，完成後請找助教檢查，經助教檢查後沒問題者請用你的學號與 Lab number 做一個檔案夾 (e.g., N96091350-Lab1)，將你的全部檔案 放入檔案夾，壓縮後上傳至課程網站(e.g.,N96091350Lab3.zip)。
4. 上傳後即可離開。
5. 未完成者可於隔日 11:55 am 前上傳至 Moodle，逾期不受理。

**Exercise 1 (20%):** Open the file “enemy.py” and complete the `draw_health_bar()` method in class `Enemy()` to show the enemy and its health bar above the enemy. Notice that the length of health bar is proportional to the instance attributes: `self.health` and `self.max_health`. Below is the sample run:

```
class Enemy:
    def __init__(self):
        self.width = 40
        self.height = 50
        self.image = pygame.transform.scale(ENEMY_IMAGE, (self.width, self.height))
        self.health = 5
        self.max_health = 10
        self.path = PATH
        self.path_pos = 0
        self.move_count = 0
        self.x = self.path[0][0]
        self.y = self.path[0][1]

    def draw(self, win):
        # draw enemy
        win.blit(self.image, (self.x - self.width // 2, self.y - self.height // 2))
        # draw enemy health bar
        self.draw_health_bar(win)

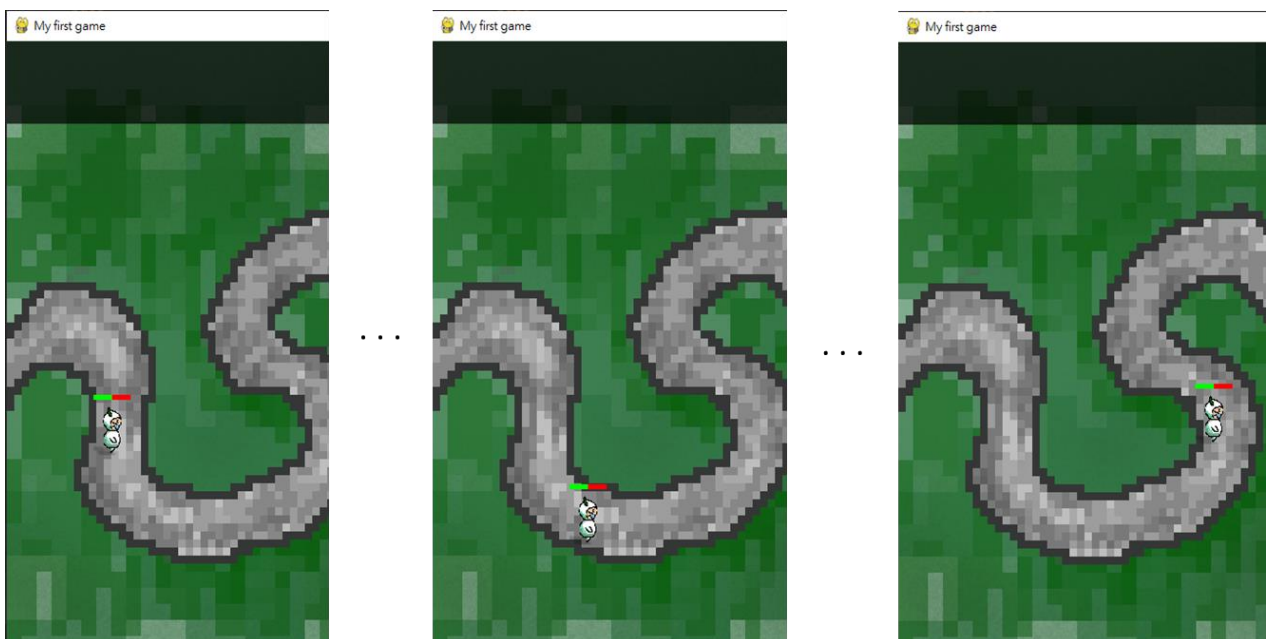
    def draw_health_bar(self, win):
        """
        Draw health bar on an enemy
        """
```



Hint:

1. Use `pygame.draw.rect(surface, color, [x, y, w, h])` to draw the bar on the win surface

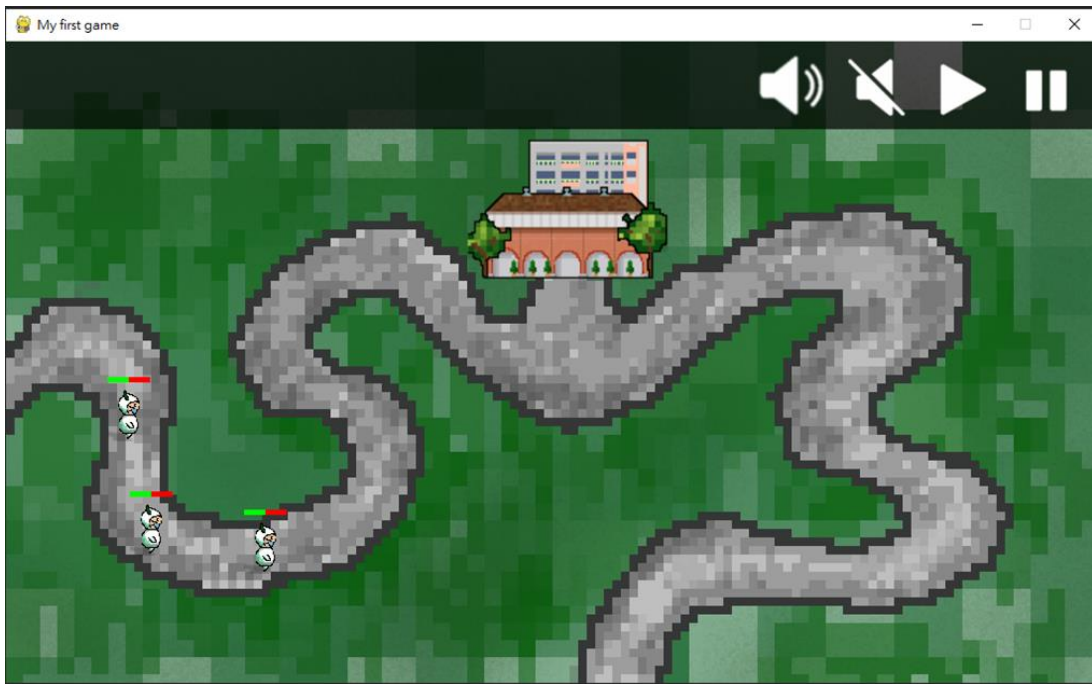
**Exercise 2 (35%):** Continued from the previous question, complete the `move()` method in class `Enemy()` to let the enemy move along the path (specifically the path point). Below is the sample run:



Hint: (The following refers to the “move\_example\_02.py”)

1. Determine how many footsteps is needed to let an enemy move from point  $i$  to point  $i+1$ .  
(For example, if the distance of the points is 12 and the stride is 1, then we need 12 footsteps to reach point  $i+1$ .)
2. Update the enemy position (i.e. `self.x`, `self.y`) in each frame.
3. Define a counter (i.e. `self.move_count`) to compute how many steps have been taken. If the counter is larger than the required footsteps, update point  $i$  and point  $i+1$  to point  $i+1$  and point  $i+2$ . Otherwise, repeat step (2) and update the counter.

**Exercise 3 (25%):** Continued from the previous question, (1) complete the `campaign` method in the class `EnemyGroup()` to let the enemies campaign in sequence in a round. Also, (2) complete the `generate` method to re-generate the enemies (when user press the “n” button). Below is the sample run:



Hint:

1. Use `append()` to add new `Enemy()`; Use `pop()` to delete `Enemy()`.

**Bonus (50%):**

Continued from the previous question, so far, we can make the enemies move along the left path. In this exercise, (1) please revise the program to let the enemy move along the right path and alternate the enemy path to the other one in the next wave.

