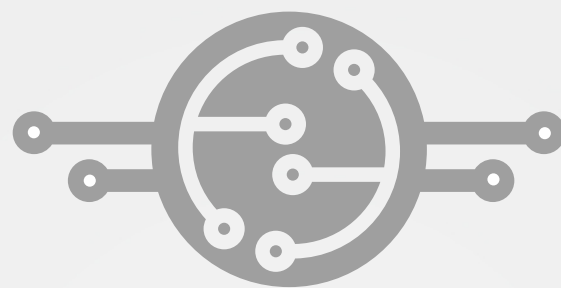


112學年_高二上學期



自主程式設計

遊戲企劃_2D賽車遊戲

方案成員:普二乙_詹正宇、普二丁_楊鈞翔

目錄:

- 摘要P.1
- 製作動機P.2
- 流程規劃P.3
- 遊戲內各程式碼介紹P.4~10
- 心得感想P.11
- 作品反思P.12

摘要：

透過多元選修課中學習到了Pygame的更多功能，我們將其整合並製作了一個有趣的小遊戲。這不僅提升了我們對程式設計的認知，更是在為我們帶來充滿創意及樂趣的體驗的同時，也挑戰自我的創造力，過程中我們不斷嘗試改變遊戲元素。完成後馬上與朋友分享，看到他們的歡笑，讓整個過程更有意義。這次冒險讓我們深刻體會到遊戲開發的許多樂趣，也激發了對創意和技術的熱忱。

製作動機：

學習 Pygame，不僅是進入遊戲開發領域的大門，更是一次挑戰自我的契機。透過本次自主學習遊戲設計，我們將更熟悉建立遊戲迴圈、處理事件、碰撞檢測等程式設計關鍵技能。而創造一個有趣的小遊戲，則是發揮創造力的最佳場所。是一次不斷挑戰自己的旅程，讓我們能夠將獨特的想法融入遊戲中，並與朋友分享快樂。這是一個激發創意、追求技術提升的精彩過程，也將是一段充滿成就感的學習之旅。

流程規劃:



2

確認好方向後
開始從各平台獲取資訊
熟悉得到的內容並加以利用

4

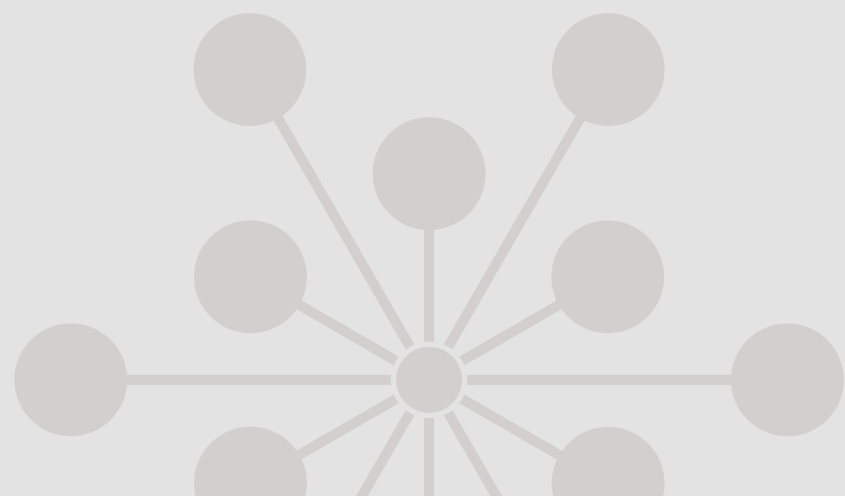
將程式套入素材
降低素材與程式不協調感
為遊戲增加細節使其更美觀

1

首先提出想要做的遊戲
比較其製作方面的難易度
選出適合且發展性高的方案

3

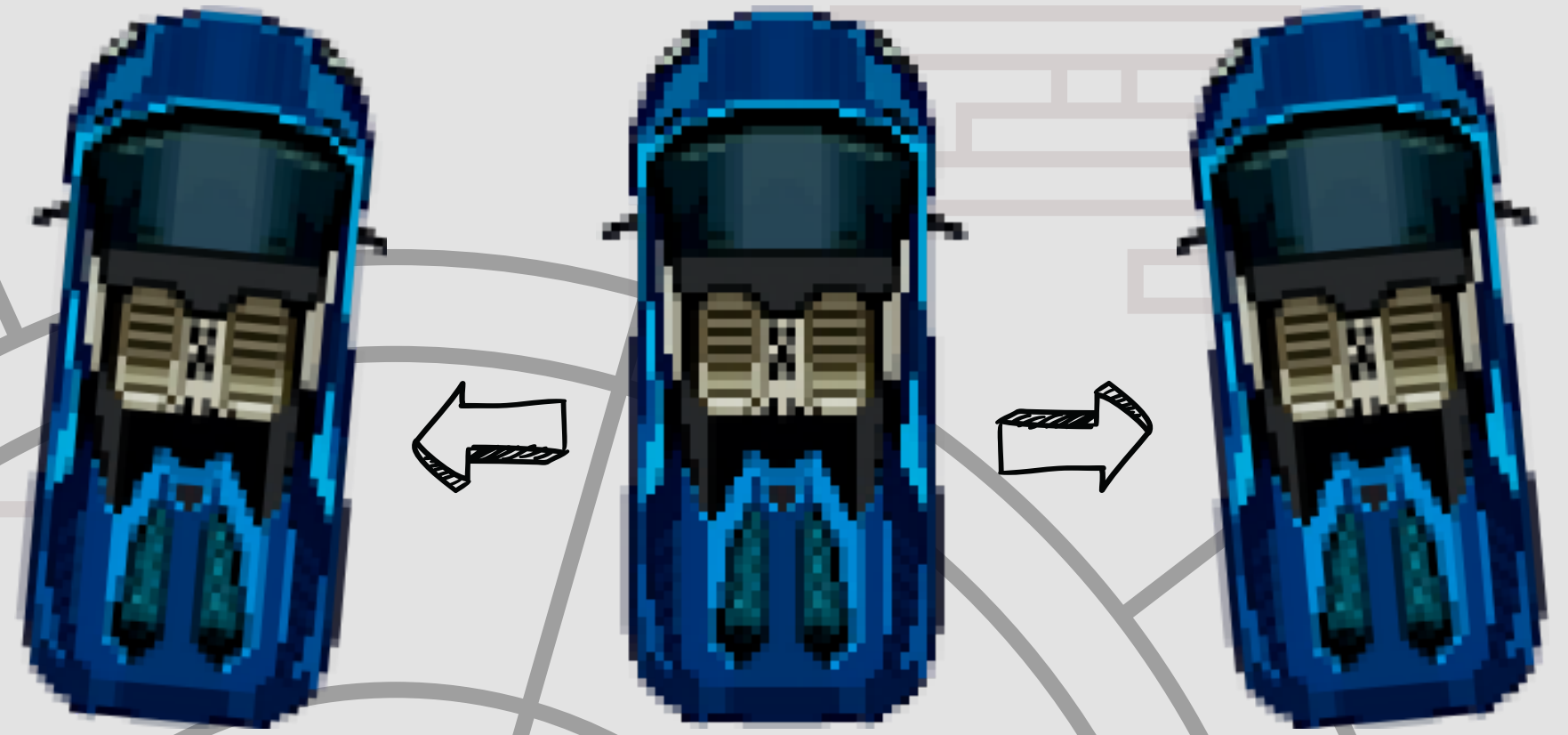
開始製造內容模板
將各自的想法呈現於程式碼中
並從其中探討程式的可執行率



創建不同檔案名稱和大小的字典



```
image_files={
    "car.png": (90, 140),
    "car_R.png": (90, 140),
    "car_L.png": (90, 140),
    "spring_road.png": None,
    "winter_road.png": None,
    "autumn_road.png": None,
    "spring_night_road.png": None,
    "winter_night_road.png": None,
    "autumn_night_road.png": None,
    "spring_tree.png": (150, 210),
    "autumn_tree.png": (150, 210),
    "winter_tree.png": (150, 210),
    "spring_night_tree.png": (150, 210),
    "autumn_night_tree.png": (150, 210),
    "winter_night_tree.png": (150, 210),
    "car2.png": (90, 140)
}
```



```
images={}
for name,size in image_files.items():
    image_path=os.path.join(dir,'resources','image',name)
    image=pygame.image.load(image_path)
    if size:
        image = pygame.transform.scale(image, size)
    images[name.split('.')[0]] = image

image_car = images["car"]
image_road = images["spring_road"]
image_Tree = images["spring_tree"]
image_car2 = images["car2"]
```

定義季節和時間的函數



```
def spring():
    global image_Tree,image_road,season
    image_Tree=images["spring_tree"]
    image_road = images["spring_road"]
    season=spring
def autumn():
    global image_Tree,image_road,season
    image_Tree=images["autumn_tree"]
    image_road = images["autumn_road"]
    season=autumn
def winter():
    global image_Tree,image_road,season
    image_Tree=images["winter_tree"]
    image_road = images["winter_road"]
    season=winter
```

```
def day():
    global image_road,image_Tree,Time
    if season==spring:
        image_road=images["spring_road"]
        image_Tree=images["spring_tree"]
    if season==winter:
        image_road=images["winter_road"]
        image_Tree=images["winter_tree"]
    if season==autumn:
        image_road=images["autumn_road"]
        image_Tree=images["autumn_tree"]
    Time=day
def night():
    global image_road,image_Tree,Time
    if season==spring:
        image_road=images["spring_night_road"]
        image_Tree=images["spring_night_tree"]
    if season==winter:
        image_road=images["winter_night_road"]
        image_Tree=images["winter_night_tree"]
    if season==autumn:
        image_road=images["autumn_night_road"]
        image_Tree=images["autumn_night_tree"]
    Time=night
```

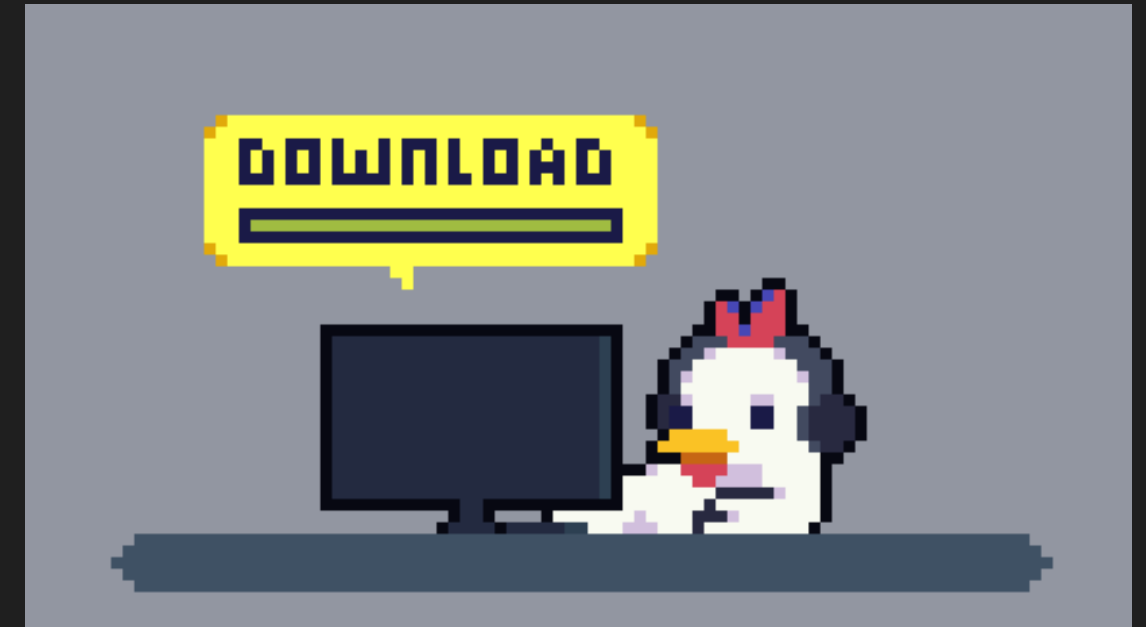
定義遊戲重置函數

```
def reset_game():
    global car_x, car_y, yline_x, yline_y, tree_left_x, tree_right_x, tree_left_y, tree_right_y
    global road_y, image_car, car2_x, car2_y, text_x, text_y, score
    boom()

    image_car=images["car"]
    car_x=800//2-car_w//2
    car_y=(900-car_h)-20
    yline_x=(800-ylines_w)//2
    yline_y=-150
    tree_left_x=random.choice(tree_left_x_path)
    tree_right_x=random.choice(tree_right_x_path)
    tree_left_y=-210
    tree_right_y=-210
    road_y=-2000
    car2_x=random.randint(225+car_w/2,575-car_w*1.5)
    car2_y=-200
    text_x=50
    text_y=200
    score=-60
```


定義開頭即結束動畫函數

```
anime=[pygame.transform.scale(pygame.image.load(os.path.join(dir,'resources','animation',f'anime ({i}).png')) ,(800,600)) for i in range(1, 32)]
def animation():
    anime_start=True
    anime_index=0
    while anime_start:
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                anime_start=False
        screen.blit(anime[anime_index], (0, 150))
        pygame.display.flip()
        clock.tick(15)
        anime_index+= 1
        if anime_index >= len(anime):
            anime_start=False
    time.sleep(0.5 )
```



```
gameover=[pygame.transform.scale(pygame.image.load(os.path.join(dir,'resources','image','gameover',f'gameover ({i}).png')) ,(800,600)) for i in range(1, 21)]
def over_animation():
    over_start=True
    over_index=0
    while over_start:
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                over_start=False
        screen.fill(black)
        screen.blit(gameover[over_index], (0, 150))
        pygame.display.flip()
        clock.tick(15)
        over_index+= 1
        if over_index >= len(gameover):
            over_start=False
    time.sleep(0.5 )
```



定義季節及時間切換函數

```
def season_change():
    global season_score
    if season_score>=1000:
        season_score=0
        if season==spring:
            autumn()
        elif season==autumn:
            winter()
        else:
            spring()

def replacement():
    global Time_score
    if Time_score>=500:
        Time_score=0
        if Time==day:
            night()
        elif Time==night:
            day()
```

在遊戲迴圈中不斷更新遊戲畫面

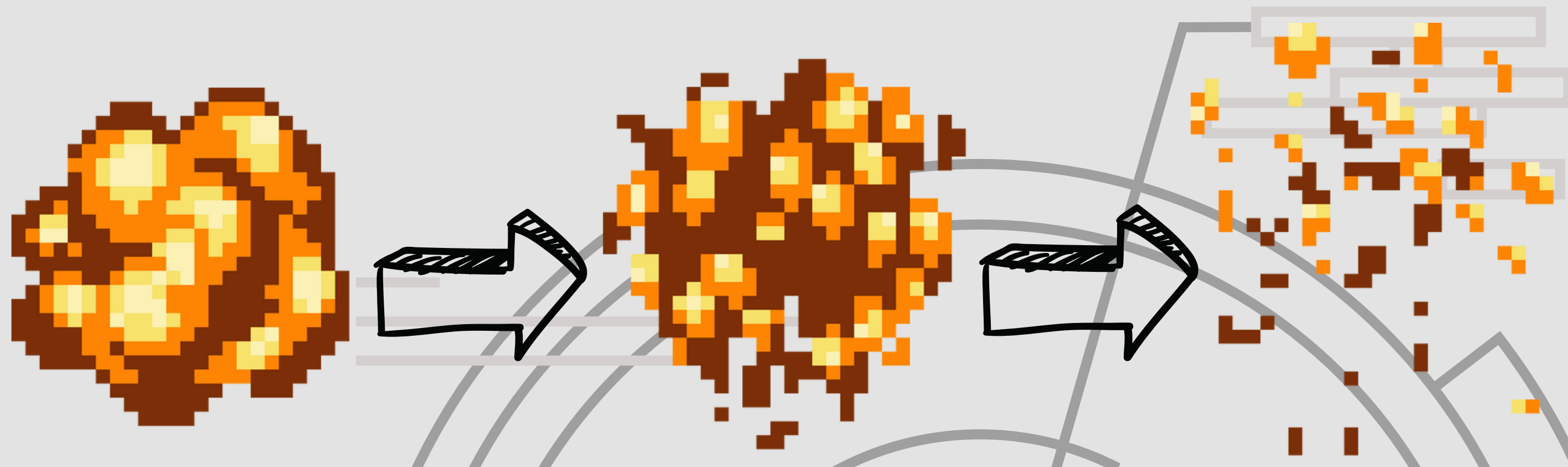
```
screen.blit(image_road,(0,road_y))
if road_y>=-2000:
    road_y+=road_speed
if road_y>0:
    road_y=-2000
pygame.draw.rect(screen,yellow,[yline_x,yline_y,yline_w,yline_h])
if yline_y>=-150:
    yline_y +=yline_speed
if yline_y>1000:
    yline_y=-150

screen.blit(image_Tree,(tree_left_x-20,tree_left_y))
if tree_left_y>=-210:
    tree_left_y+=yline_speed
if tree_left_y>=random.randint(1000,10000):
    tree_left_x=random.choice(tree_left_x_path)
    tree_left_y=-210

screen.blit(image_Tree,(tree_right_x-20,tree_right_y))
if tree_right_y>=-210:
    tree_right_y+=yline_speed
if tree_right_y>=random.randint(1000,10000):
    tree_right_x=random.choice(tree_right_x_path)
    tree_right_y=-210

screen.blit(image_car2,(car2_x,car2_y))
if car2_y >= -200:
    car2_y += car2_speed
if car2_y >= random.randint(1150,5000):
    car2_x = random.randint(225+car_w/2,575-car_w*1.5)
    car2_y =-200
```

定義車輛撞擊時的爆炸效果



```
Boom=[pygame.transform.scale(pygame.image.load(os.path.join(dir,'resources','image',f'bone({i}).png')) ,(250,250)) for i in range(1,8)]
def boom():
    boom_start=True
    boom_index=0
    while boom_start:
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                boom_start=False
        screen.blit(Boom[boom_index], (car_x-70, car_y-70))
        pygame.display.flip()
        clock.tick(5)
        boom_index+= 1
        if boom_index >= len(Boom):
            boom_start=False
```

定義車輛與障礙物的碰撞範圍

```
if car_x<=tree_left_x+75 and car_x+car_w>=tree_left_x and car_y<=tree_left_y+170 and car_y+car_h>tree_left_y:
    screen.blit(text,(text_x,text_y))
    stop=True
    if text_y<=1000:
        text_y+=2
    if text_y>=1000:
        game_over=True
if car_x<=tree_right_x+70 and car_x+car_w>=tree_right_x+40 and car_y<=tree_right_y+170 and car_y+car_h>tree_right_y:
    screen.blit(text,(text_x,text_y))
    stop=True
    if text_y<=1000:
        text_y+=2
    if text_y>=1000:
        game_over=True
elif car_x<=car2_x+75 and car_x+car_w>=car2_x+20 and car_y<=car2_y+120 and car_y+car_h>car2_y:
    screen.blit(text,(text_x,text_y))
    stop=True
    if text_y<=1000:
        text_y+=2
    if text_y>=1000:
        game_over=True
```


心得感想：

我們可以完全自己決定遊戲的內容，充分發揮自己的創意。不只可以學技術，還可以將其作為藝術的一種表達方式，讓我們有機會在遊戲中盡情展現出自己的風格。

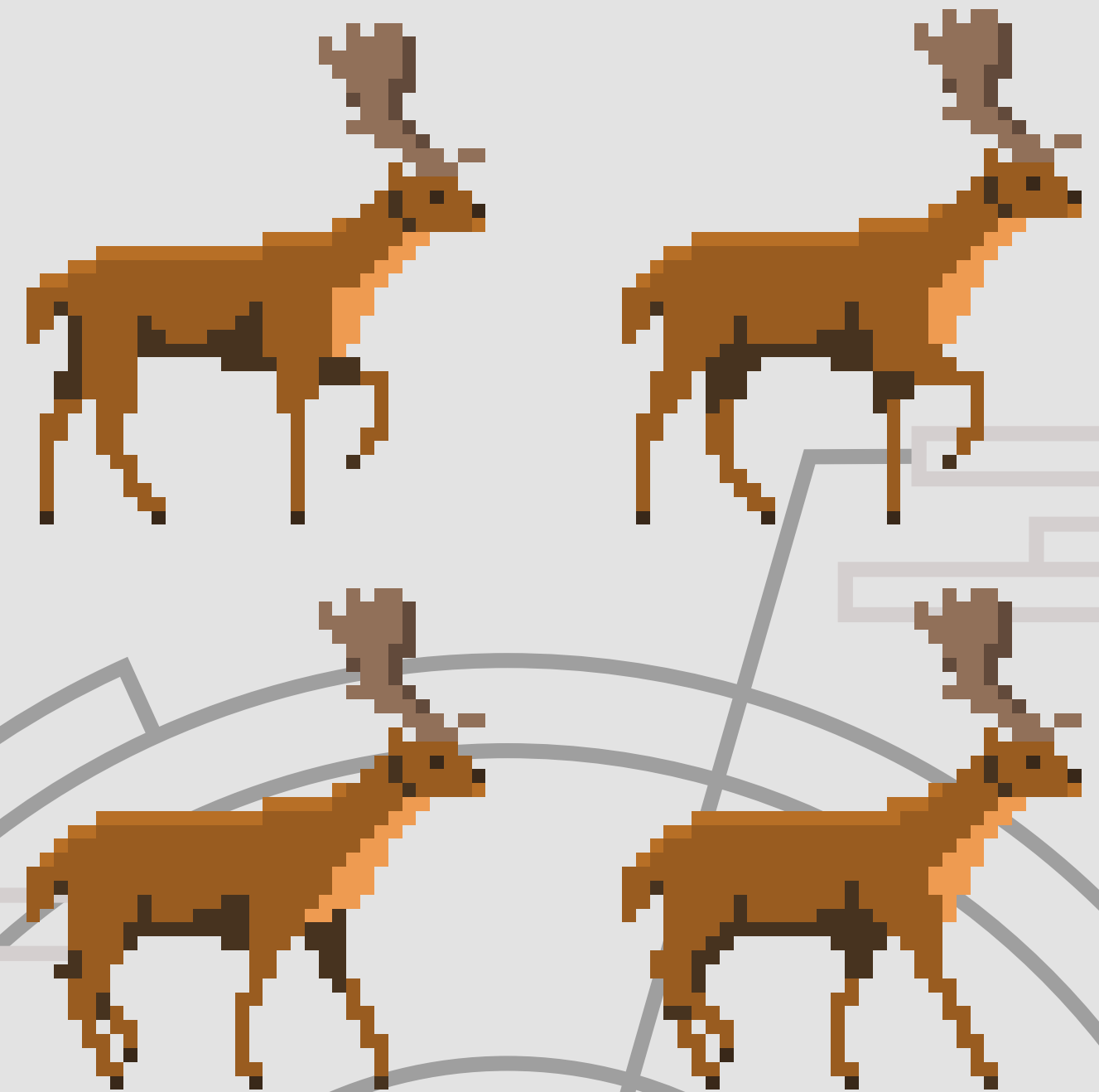
挑戰自己的創意是最有意思的部分。每次修改遊戲的樣子和玩法都是在思考和嘗試新點子。這讓我們更了解自己的想法也可以避免日後犯同樣的錯誤，讓我們對遊戲開發更有熱情。在製作的過程中也常常在討論該如何用程式把我們想要的東西製作出來，有時會突然靈機一動，把突然想到的程式碼寫在筆記本上，等回到家的時候繼續研究，日積月累下，已有多次在技術上突破，對於程式的理解也愈發精進，進步的同時也充滿了熱情。

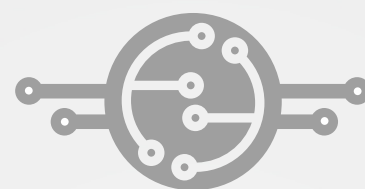


作品反思：

雖然我們製作的小遊戲無法與市面上的遊戲相媲美，但我們仍在學習的過程中。遊戲中可能存在一些比較初階的技術和操作，甚至可能會有一些導致bug的錯誤。但整體而言，這是一款能夠分享給朋友消磨時間的娛樂遊戲。當然，我們會持續提升技術水平，改進遊戲，以使其擁有更樣的玩法，同時減少可能出現bug的機會。

想加入遊戲中的鹿





感謝您的閱讀

報告到此結束