c0b230076b 追加機能4完成 cd8826f · 2 minutes ago

205 lines (178 loc) · 8.24 KB

```python
import pygame
import sys

# Pygameの初期化
pygame.init()

# 画面サイズとマス目サイズの設定
SCREEN_WIDTH = 800  # 画面の幅
SCREEN_HEIGHT = 600  # 画面の高さ
GRID_SIZE = 80  # 1つのマスのサイズ
INFO_AREA_HEIGHT = 80  # 上部の情報エリアの高さ
game_start = False  # ゲームが開始されているかの真理値

# 色の定義（RGB形式）
GREEN = (0, 128, 0)  # 背景の緑色
WHITE = (255, 255, 255)  # マス目の線の色
BLACK = (0, 0, 0)  # テキストの色
GRAY = (200, 200, 200)  # 情報エリアの背景色
RED = (255, 0, 0)  # ゾンビの色
BLUE = (0, 0, 255)  # 植物の色


# 画面の作成
screen = pygame.display.set_mode((SCREEN_WIDTH, SCREEN_HEIGHT))
pygame.display.set_caption("Plants vs Zombies 風ゲーム")

# フォントの設定
font = pygame.font.Font(None, 36)

# ゾンビクラスの定義
class Zombie:
    def __init__(self, x, y, speed):
        self.rect = pygame.Rect(x, y, GRID_SIZE, GRID_SIZE)  # ゾンビを長方形で表す
        self.speed = speed
        self.alive = True  # 障害物に到達すると停止

    def move(self, obstacles):
        if self.alive:
            # ゾンビが障害物に衝突しているか確認
            for obstacle in obstacles:
                if self.rect.colliderect(obstacle):
```

```
42                      self.alive = False  # 衝突したら停止
43                      return
44                  # 左に移動
45                  self.rect.x -= self.speed
46
47          def draw(self, surface):
48              pygame.draw.rect(surface, RED, self.rect)
49
50      def draw_title(screen: pygame.Surface):
51          """
52          タイトル画面を表示する関数
53          引数1 screen：画面Surface
54          """
55          title = pygame.Surface((SCREEN_WIDTH, SCREEN_HEIGHT))  # タイトル画面の背景Surface生成
56          pygame.draw.rect(title, (230,230,250), pygame.Rect(0,0,SCREEN_WIDTH,SCREEN_HEIGHT))
57          fonto = pygame.font.SysFont("hgp創英角ﾎﾟｯﾌﾟ体", 50)  # タイトルの文字Surface生成
58          title_txt = fonto.render("こうかとん VS ゾンビ", True, (0, 0, 0))  #文字をこうかとん VSゾンビ,色を黒
59          title_txt_rct = title_txt.get_rect()  #タイトルテキストのrectを抽出
60          title_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2-100)
61          dis_txt = fonto.render("Enterを押してゲームスタート", True, (0,0,0))  # 説明の文字Surface生成
62          dis_txt_rct = dis_txt.get_rect()  # 説明テキストのrectを抽出
63          dis_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2+50)
64          kk_img = pygame.transform.rotozoom(pygame.image.load("ex5/fig/2.png"), 0, 1.5)
65          kk_rct = kk_img.get_rect()
66          kk_rct.center = 300, 100
67          screen.blit(title, [0,0])
68          screen.blit(title_txt, title_txt_rct)
69          screen.blit(dis_txt, dis_txt_rct)
70          screen.blit(kk_img, kk_rct)
71
72      # テキストを描画する関数
73      def draw_text(surface, text, x, y, color):
74          rendered_text = font.render(text, True, color)
75          surface.blit(rendered_text, (x, y))
76
77      # マス目を描画する関数
78      def draw_grid(surface, width, height, grid_size, offset_y):
79          for x in range(0, width, grid_size):
80              pygame.draw.line(surface, WHITE, (x, offset_y), (x, height))
81          for y in range(offset_y, height, grid_size):
82              pygame.draw.line(surface, WHITE, (0, y), (width, y))
83
84      # 情報エリアを描画する関数
85      def draw_info_area(surface, width, height):
86          pygame.draw.rect(surface, GRAY, (0, 0, width, height))
87          draw_text(surface, "score: 0", 20, 20, BLACK)
88          draw_text(surface, "set", 200, 20, BLACK)
89
90      def draw_finish(screen: pygame.Surface):
91          """
92          クリア画面を表示する関数
93          引数1 screen：画面Surface
94          """
95          clear = pygame.Surface((SCREEN_WIDTH, SCREEN_HEIGHT))
96          pygame.draw.rect(clear, (230,230,0), pygame.Rect(0,0,SCREEN_WIDTH,SCREEN_HEIGHT))
97          fonto = pygame.font.SysFont("hgp創英角ﾎﾟｯﾌﾟ体", 50)
98          title_txt = fonto.render("クリアおめでとう", True, (0, 0, 0))  #文字をクリアおめでとう,色を黒に設定
99          title_txt_rct = title_txt.get_rect()  #タイトルテキストのrectを抽出
```

```
        title_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2-100)
        dis_txt = fonto.render("×を押して終了してね", True, (0,0,0))
        dis_txt_rct = dis_txt.get_rect()
        dis_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2+50)
        kk_img = pygame.transform.rotozoom(pygame.image.load("ex5/fig/9.png"), 0, 2)
        kk_rct = kk_img.get_rect()
        kk_rct.center = 500, 500
        screen.blit(clear, [0,0])
        screen.blit(title_txt, title_txt_rct)
        screen.blit(dis_txt, dis_txt_rct)
        screen.blit(kk_img, kk_rct)

    def draw_gameover(screen: pygame.Surface):
        """
        ゲームオーバー画面を表示する関数
        引数1 screen：画面Surface
        """
        gameover = pygame.Surface((SCREEN_WIDTH, SCREEN_HEIGHT))
        pygame.draw.rect(gameover, (0,0,0), pygame.Rect(0,0,SCREEN_WIDTH,SCREEN_HEIGHT))
        fonto = pygame.font.SysFont("hgp創英角ﾎﾟｯﾌﾟ体", 100)
        dis = pygame.font.SysFont("hgp創英角ﾎﾟｯﾌﾟ体", 50)
        title_txt = fonto.render("Game Over", True, (255, 255, 255))  #文字をGame Over,色を白に設定
        title_txt_rct = title_txt.get_rect()  #タイトルテキストのrectを抽出
        title_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2)
        dis_txt = dis.render("×を押して終了してね", True, (255,255,255))
        dis_txt_rct = dis_txt.get_rect()
        dis_txt_rct.center = (SCREEN_WIDTH/2, SCREEN_HEIGHT/2+100)
        kk_img = pygame.transform.rotozoom(pygame.image.load("ex5/fig/8.png"), 0, 2)
        kk_rct = kk_img.get_rect()
        kk_rct.center = 500, 500
        screen.blit(gameover, [0,0])
        screen.blit(title_txt, title_txt_rct)
        screen.blit(dis_txt, dis_txt_rct)
        screen.blit(kk_img, kk_rct)


    # メインのゲームループ
    def main():
        global game_start
        clock = pygame.time.Clock()

        # ゾンビを1体生成
        zombie = Zombie(SCREEN_WIDTH, INFO_AREA_HEIGHT + GRID_SIZE * 2, 2)

        # 障害物（植物）を格納するリスト
        plants = []

        # ゲームループ
        while True:
            for event in pygame.event.get():
                if event.type == pygame.QUIT:
                    pygame.quit()
                    sys.exit()
                elif event.type == pygame.KEYDOWN and event.key == pygame.K_RETURN:
                    game_start = True
                # elif event.type == pygame.MOUSEBUTTONDOWN:
                #     # マウスクリックで植物を配置
                #     mouse_x, mouse_y = event.pos
```

```
158                 #     if mouse_y > INFO_AREA_HEIGHT:  # 情報エリア以外をクリック可能
159                 #         grid_x = (mouse_x // GRID_SIZE) * GRID_SIZE
160                 #         grid_y = (mouse_y // GRID_SIZE) * GRID_SIZE
161                 #         plant_rect = pygame.Rect(grid_x, grid_y, GRID_SIZE, GRID_SIZE)
162                 #         plants.append(plant_rect)
163 
164             if game_start == False:
165                 draw_title(screen)
166                 pygame.display.update()
167             else:
168                 if event.type == pygame.MOUSEBUTTONDOWN:
169                     # マウスクリックで植物を配置
170                     mouse_x, mouse_y = event.pos
171                     if mouse_y > INFO_AREA_HEIGHT:  # 情報エリア以外をクリック可能
172                         grid_x = (mouse_x // GRID_SIZE) * GRID_SIZE
173                         grid_y = (mouse_y // GRID_SIZE) * GRID_SIZE
174                         plant_rect = pygame.Rect(grid_x, grid_y, GRID_SIZE, GRID_SIZE)
175                         plants.append(plant_rect)
176                 # 背景の描画
177                 screen.fill(GREEN)
178 
179                 # 情報エリアの描画
180                 draw_info_area(screen, SCREEN_WIDTH, INFO_AREA_HEIGHT)
181 
182                 # マス目の描画
183                 draw_grid(screen, SCREEN_WIDTH, SCREEN_HEIGHT, GRID_SIZE, INFO_AREA_HEIGHT)
184 
185                 # 植物の描画
186                 for plant in plants:
187                     pygame.draw.rect(screen, BLUE, plant)
188 
189                 # ゾンビの動きと描画
190                 zombie.move(plants)
191                 zombie.draw(screen)
192 
193                 # クリアしたらクリア画面の表示
194                 if len(plants) > 100:
195                     draw_finish(screen)
196                 # ゲームオーバーになるとゲームオーバー画面の表示
197                 if zombie.rect.x <= 0:
198                     draw_gameover(screen)
199                 # 画面の更新
200                 pygame.display.update()
201                 clock.tick(60)
202 
203     # メイン関数の実行
204     if __name__ == "__main__":
205         main()
```