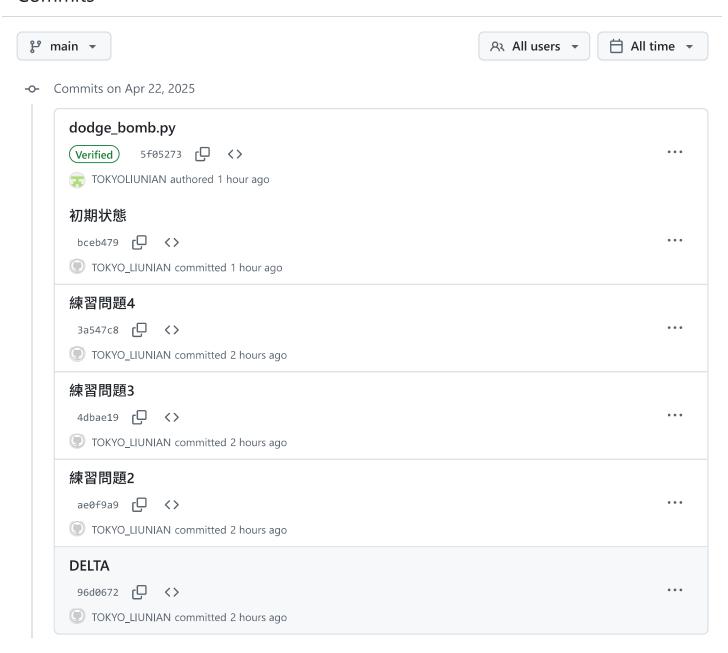
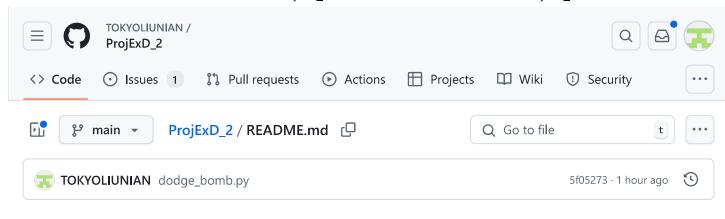


Commits





89 lines (73 loc) · 2.33 KB

```
Raw [ L
                                                                              83
                                                                                                            \langle \rangle
Preview
           Code
                    Blame
          import os
    1
    2
          import random
    3
          import sys
    4
          import pygame as pg
    6
    7
          WIDTH, HEIGHT = 1100, 650
          DELTA = {
    8
    9
              pg.K_UP: (0, -5),
   10
              pg.K_DOWN: (0, +5),
   11
              pg.K_LEFT: (-5, 0),
              pg.K_RIGHT: (+5, 0),
   12
          }
   13
   14
          os.chdir(os.path.dirname(os.path.abspath(__file__)))
   15
   16
          def check_bound(rct: pg.Rect) -> tuple[bool, bool]:
   17
   18
              yoko, tate = True, True # 横,縦方向用の変数
   19
   20
              # 横方向判定
   21
              if rct.left < 0 or WIDTH < rct.right: # 画面外だったら
   22
                  yoko = False
              # 縦方向判定
   23
              if rct.top < 0 or HEIGHT < rct.bottom: # 画面外だったら
   24
                  tate = False
   25
   26
              return yoko, tate
   27
   28
   29
   30
   31
          def main():
              pg.display.set_caption("逃げろ! こうかとん")
   32
   33
              screen = pg.display.set_mode((WIDTH, HEIGHT))
              # こうかとん初期化
   34
              bg_img = pg.image.load("fig/pg_bg.jpg")
   35
              kk_img = pg.transform.rotozoom(pg.image.load("fig/3.png"), 0, 0.9)
   36
              kk_rct = kk_img.get_rect()
   37
   38
              kk_rct.center = 300, 200
              # 爆弾初期化
   39
              bb_img = pg.Surface((20, 20))
   40
              pg.draw.circle(bb_img, (255, 0, 0), (10, 10), 10)
   41
   42
              bb_rct = bb_img.get_rect()
```

```
bb_rct.centerx = random.randint(0, WIDTH)
43
44
           bb_rct.centery = random.randint(0, HEIGHT)
45
           bb img.set colorkey((0, 0, 0))
           vx, vy = +5, +5
46
47
           clock = pg.time.Clock()
48
           tmr = 0
49
           while True:
50
               for event in pg.event.get():
51
                   if event.type == pg.QUIT:
53
                       return
               screen.blit(bg_img, [0, 0])
54
55
               # こうかとん
56
57
               if kk_rct.colliderect(bb_rct):
                   print("Game Over")
58
                   return
59
60
               key_lst = pg.key.get_pressed()
61
62
               sum mv = [0, 0]
63
               for key, mv in DELTA.items():
64
65
                   if key_lst[key]:
                       sum_mv[0] += mv[0] # 左右方向
66
                       sum_mv[1] += mv[1] # 上下方向
67
68
69
               kk_rct.move_ip(sum_mv)
70
               if check_bound(kk_rct) != (True, True): # 画面外だったら
                   kk_rct.move_ip(-sum_mv[0], -sum_mv[1]) # 画面内に戻す
71
               screen.blit(kk_img, kk_rct)
72
               bb_rct.move_ip(vx, vy) # 爆弾の移動
73
74
               yoko, tate = check_bound(bb_rct)
75
               if not yoko: # 左右
                   vx *= -1
76
               if not tate: # 上下
77
                   vy *= -1
78
79
               screen.blit(bb img, bb rct) # 爆弾の描画
80
               pg.display.update()
               tmr += 1
81
               clock.tick(50)
82
83
84
85
       if __name__ == "__main__":
86
           pg.init()
           main()
87
           pg.quit()
88
89
           sys.exit()
```

