



EXEMPLO CONSOLIDAÇÃO GAME JAM DA PONTUAÇÃO E CÓDIGO PARCIAL DO GAME PAC-MAN

```
import pygame

pygame.init()

screen = pygame.display.set_mode((800, 600), 0)

font = pygame.font.SysFont("arial", 20, True, False)

AMARELO = (255, 255, 0)
PRETO = (0, 0, 0)
AZUL = (0, 0, 255)
VELOCIDADE = 1
```

```
class Cenario:
    def __init__(self, tamanho, pac):
        self.pacman = pac
        self.pontos = 0
        self.tamanho = tamanho
        self.matriz = [
            [2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2],
            [2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2],
            [2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2],
            [2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2],
            [2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2],
            [2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2],
            [2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2],
            [2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 2],
```



```
[2, 1, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2],  
[2, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 2],  
[2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2],  
[2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2],  
[2, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 0, 0, 0, 0, 2, 2, 1, 2, 2, 1, 2, 2, 2, 1, 2, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 0, 0, 0, 0, 0, 0, 2, 1, 2, 2, 1, 2, 2, 2, 1, 2],  
[2, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2, 0, 0, 0, 0, 0, 0, 2, 1, 2, 2, 1, 1, 1, 1, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 0, 0, 0, 0, 0, 2, 1, 2, 2, 1, 2, 2, 2, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2],  
[2, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2],  
[2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2],  
[2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2],  
[2, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2],  
[2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 1, 2],  
[2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 1, 2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 2],  
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
```

```
]
```

```
def pintar_pontos(self, tela):  
    pontos_x = self.tamanho * 30  
    pontos_img = font.render("Score {}".format(self.pontos), True, AMARELO)  
    tela.blit(pontos_img, (pontos_x, 50))
```



```
def pintar_linha(self, tela, numero_linha, linha):  
    for numero_coluna, coluna in enumerate(linha):  
        x = numero_coluna * self.tamanho  
        y = numero_linha * self.tamanho  
        half = self.tamanho // 2  
        cor = PRETO  
        if coluna == 2:  
            cor = AZUL  
        pygame.draw.rect(tela, cor, (x, y, self.tamanho, self.tamanho), 0)  
        if coluna == 1:  
            pygame.draw.circle(tela, AMARELO, (x + half, y + half),  
                               self.tamanho // 10, 0)  
  
def pintar(self, tela):  
    for numero_linha, linha in enumerate(self.matriz):  
        self.pintar_linha(tela, numero_linha, linha)  
self.pintar_pontos(tela)  
  
def calcular_regras(self):  
    col = self.pacman.coluna_intencao  
    lin = self.pacman.linha_intencao  
    if 0 <= col < 28 and 0 <= lin < 29:  
        if self.matriz[lin][col] != 2:  
            self.pacman.aceitar_movimento()  
            if self.matriz[lin][col] == 1:  
                self.pontos += 1  
                self.matriz[lin][col] = 0  
print(self.pontos)
```



```
class Pacman:  
    def __init__(self, tamanho):  
        self.coluna = 1  
        self.linha = 1  
        self.centro_x = 400  
        self.centro_y = 300  
        self.tamanho = tamanho  
        self.vel_x = 0  
        self.vel_y = 0  
        self.raio = self.tamanho // 2  
        self.coluna_intencao = self.coluna  
        self.linha_intencao = self.linha  
  
    def calcular_regras(self):  
        self.coluna_intencao = self.coluna + self.vel_x  
        self.linha_intencao = self.linha + self.vel_y  
        self.centro_x = int(self.coluna * self.tamanho + self.raio)  
        self.centro_y = int(self.linha * self.tamanho + self.raio)  
  
    def pintar(self, tela):  
        # Desenhar o corpo do Pacman  
        pygame.draw.circle(tela, AMARELO, (self.centro_x, self.centro_y), self.raio, 0)  
  
        # Desenho da boca do Pacman  
        canto_boca = (self.centro_x, self.centro_y)  
        labio_superior = (self.centro_x + self.raio, self.centro_y - self.raio)  
        labio_inferior = (self.centro_x + self.raio, self.centro_y)
```



```
pontos = [canto_boca, labio_superior, labio_inferior]

pygame.draw.polygon(tela, PRETO, pontos, 0)

# Olho do Pacman

olho_x = int(self.centro_x + self.raio / 3)

olho_y = int(self.centro_y - self.raio * 0.70)

olho_raio = int(self.raio / 10)

pygame.draw.circle(tela, PRETO, (olho_x, olho_y), olho_raio, 0)

def processar_eventos(self, eventos):

    for e in eventos:

        if e.type == pygame.KEYDOWN:

            if e.key == pygame.K_RIGHT:

                self.vel_x = VELOCIDADE

            elif e.key == pygame.K_LEFT:

                self.vel_x = -VELOCIDADE

            elif e.key == pygame.K_UP:

                self.vel_y = -VELOCIDADE

            elif e.key == pygame.K_DOWN:

                self.vel_y = VELOCIDADE

        elif e.type == pygame.KEYUP:

            if e.key == pygame.K_RIGHT:

                self.vel_x = 0

            elif e.key == pygame.K_LEFT:

                self.vel_x = 0

            elif e.key == pygame.K_UP:

                self.vel_y = 0

            elif e.key == pygame.K_DOWN:

                self.vel_y = 0
```



```
def aceitar_movimento(self):
    self.linha = self.linha_intencao
    self.coluna = self.coluna_intencao

if __name__ == "__main__":
    size = 600 // 30
    pacman = Pacman(size)
    cenario = Cenario(size, pacman)

while True:
    # Calcular as regras
    pacman.calcular_regras()
    cenario.calcular_regras()

    # Pintar a tela
    screen.fill(PRETO)
    cenario.pintar(screen)
    pacman.pintar(screen)
    pygame.display.update()
    pygame.time.delay(100)

    # Captura os eventos
    eventos = pygame.event.get()

    for e in eventos:
        if e.type == pygame.QUIT:
            exit()

        pacman.processar_eventos(eventos)
```