Universidade Federal Fluminense
Instituto de Computação
TCC00291 - Computação Gráfica
Profª.: Aura Conci

Niterói, 8 de Dezembro de 2018

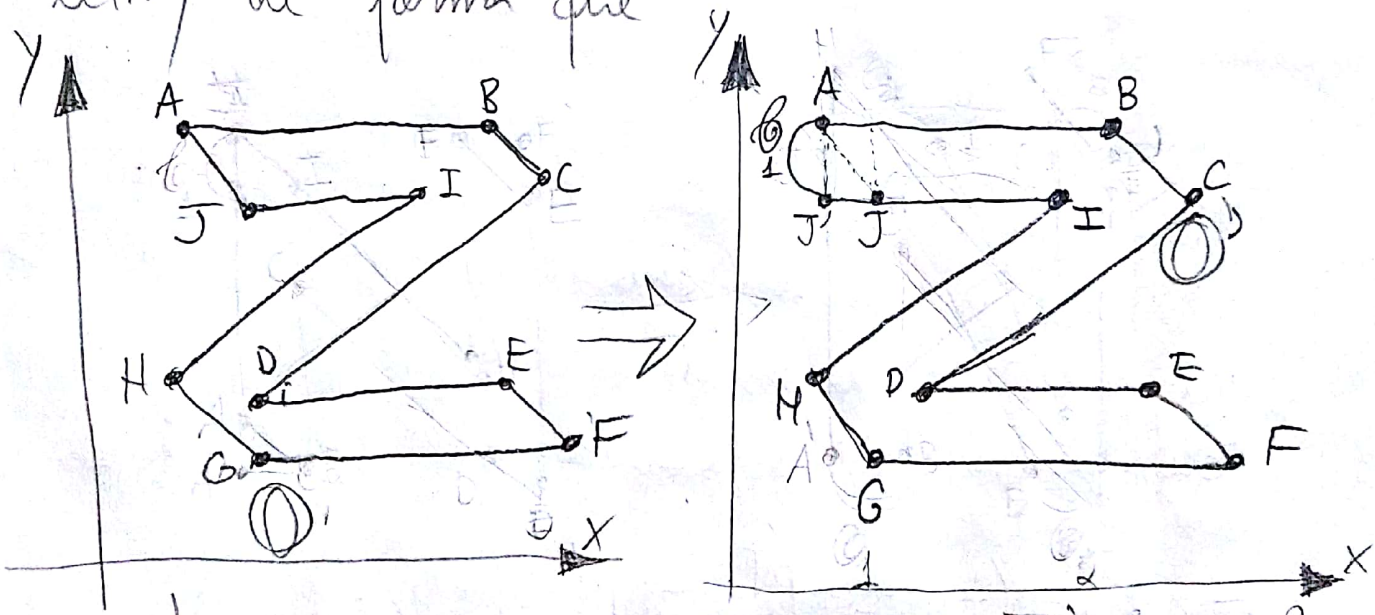
Felipe Genú Simões - 116031072
Pedro Paulo Bastos Teixeira - 615031106

Trabalho 3 - Relatório

Será, a exemplo do trabalho 2, utilizada a mesma letra do trabalho referido, conforme acordado com a Profª: Aura, a letra Z

Dessa forma, temos que...

Apenas será mudada a estrutura de dados, da letra, de forma que



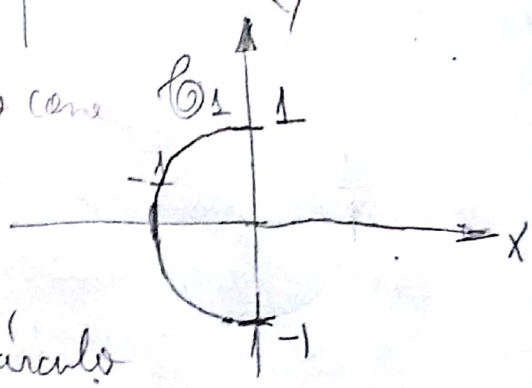
$$\mathcal{O}' = \{\overline{AB}, \overline{BC}, \overline{CD}, \overline{DE}, \overline{EF}, \overline{FG}, \overline{GH}, \overline{HI}, \overline{IJ}, \mathcal{C}_1\}$$

Onde $\mathcal{C}_1$, é uma curva, definida por $\mathcal{C}_1 = \overline{J'A}$

Essa curva na verdade é um semicírculo com concavidade para a direita.

O semicírculo será definido parametricamente.

Um semicírculo côncavo à direita, qualquer, é definido como



$$\mathcal{C}_1 = \mathcal{P}(t) = (-h \cdot |\cos t|,\ h \cdot \text{sen}\, t)$$

onde h é a excentricidade do semicírculo

Assim, na letra, será aplicado sobre o ponto A e o semicírculo será gerado em vários pontos.

ele será gerado por uma função que recebe ...

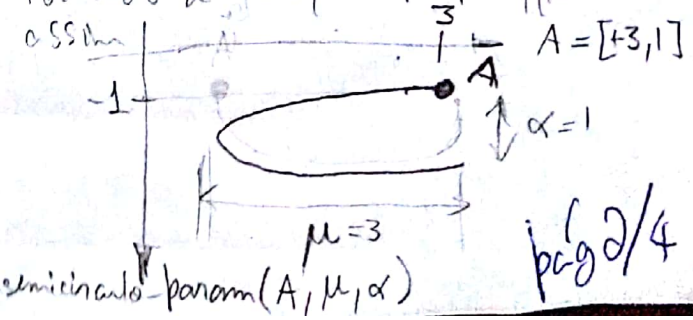
$P_i = [x, y]$ o ponto inicial de aplicação do semicírculo (côncavo); h_y: a excentricidade vertical do semicírculo

h_x: a excentricidade lateral do semicírculo

```
def desenha_semicirculo_param(Pi, hx, hy):
    Pu = (-hx*|cos t| + Pix; hy*sen t + Piy)
    for t in range(0, 2π):
        pygame.desenha_ponto((-hx|cos t| + Pix; hy sen t + Piy))
```

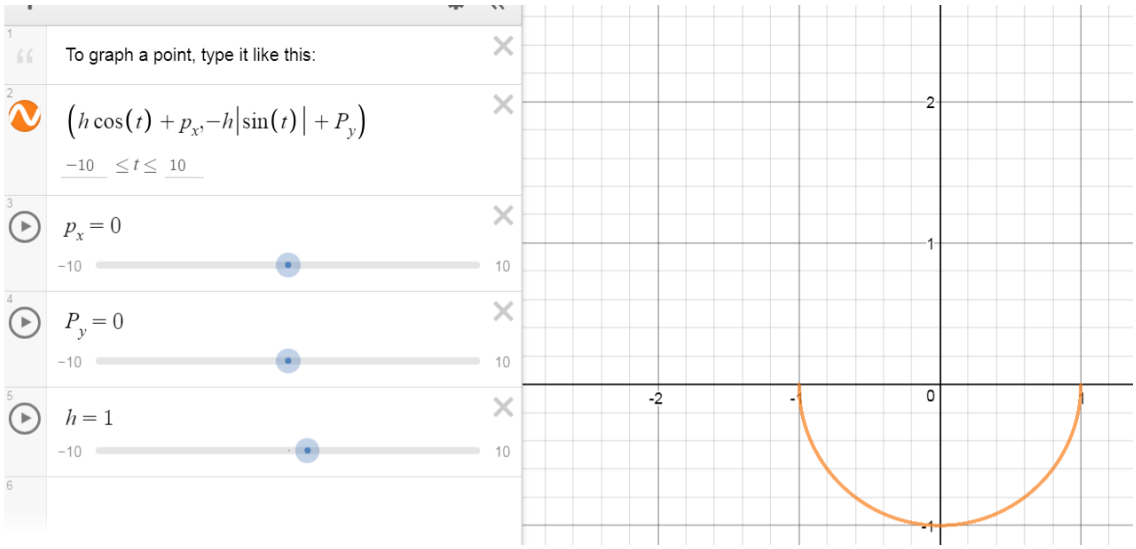
(prendo código)

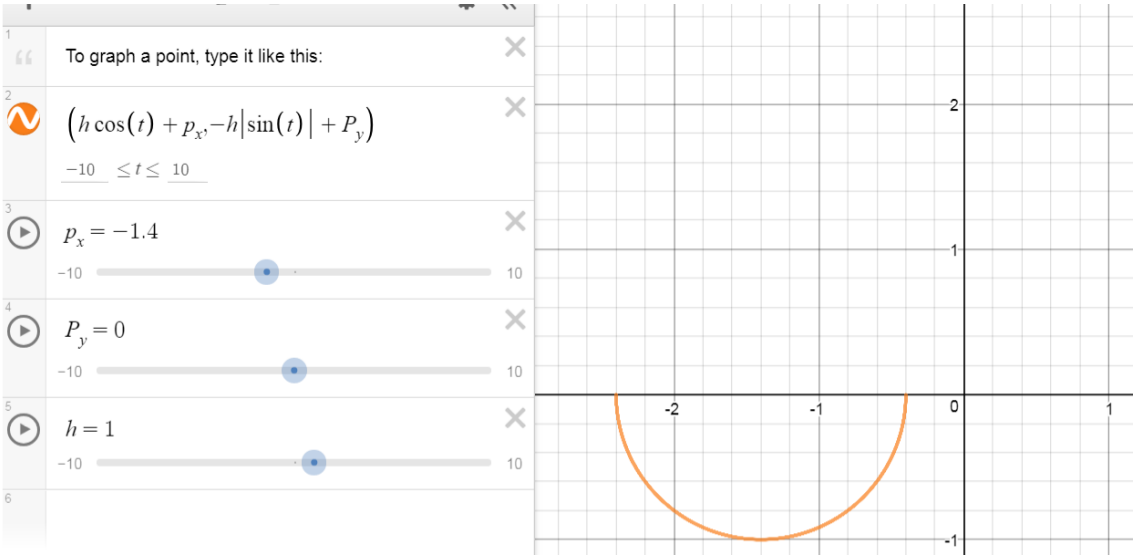
rodando a função pós-plicada assim

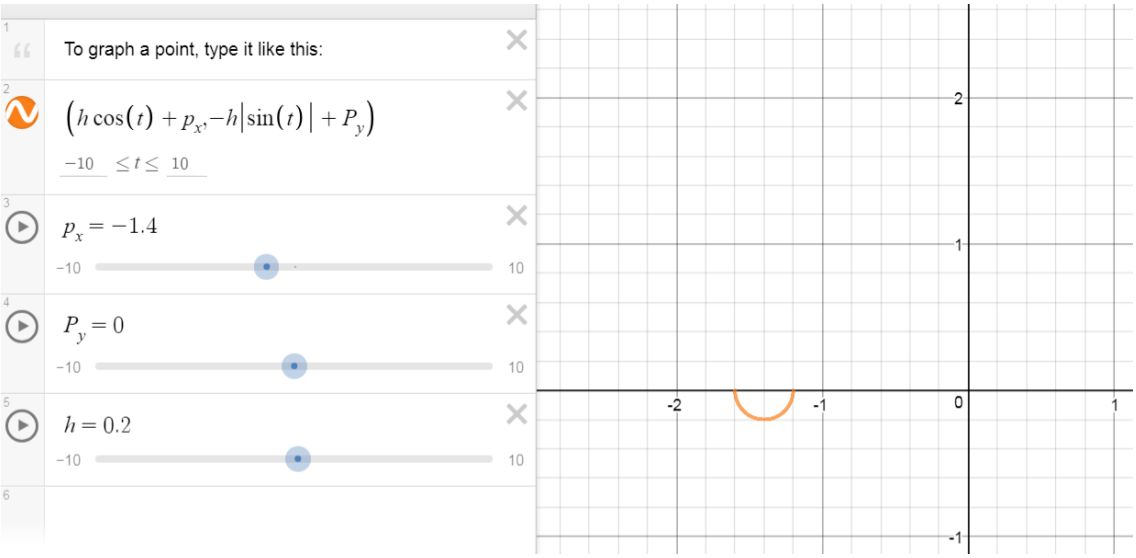


desenha_semicirculo_param(A, μ, α)

Ou seja, para chegar ao tamanho certo (encaixando na letra), vai se variando a excentricidade do semicírculo, h_x e h_y para variar o tamanho(largura e altura do semicirculo).
e vai se variando P_i, que é a âncora da parábola, para se transladar ate a posição desejada







```
def desenha_semicirculo_pygame(P_i, h_x, h_y, tela, cor, width):
    theta = 0
    t = radians(theta)
    while t <= radians(360):
        desenha_pygame_ponto([-h_x*cos(t)+P_i[0], -h_y*sin t
        theta = theta + radians(0.01)
        t = t + theta
```

```
def desenha_pygame_ponto(P, tela, cor, width):
    pygame.draw.line(tela, P, P, tela, cor, width)
```

(em Python)