

Trab 4 - Octave

Anny Caroline Correa Chagas
Ciência da Computação, UERJ

21 de Julho de 2019

Primeira parte

1. Crie uma função no Octave que realize a rotação de um objeto geométrico sobre o seu centro.
2. A função deve ter como parâmetros o ângulo de rotação (em graus) e uma matriz com os pontos que representam os vértices do objeto (o objeto pode ter um número arbitrário de pontos). Cada ponto deve corresponder a uma coluna da matriz, como por exemplo: $\begin{bmatrix} 0 & 0 & 1 & 1; 0 & 1 & 1 & 0; 1 & 1 & 1 & 1 \end{bmatrix}$;
A matriz acima representa o quadrado unitário com o primeiro vértice na origem.
3. Mostre em figuras diferentes o objeto original e o objeto rotacionado.
4. Considere utilizar as seguintes funções do Octave: `axis`, `hold on`, `line`; e lembre-se de que as funções trigonométricas têm como entrada ângulos em radianos.
5. Submeta o código da função e um script que chame a função (em arquivos `.m`).

Resposta:

```
1 function retval = desenhaPoligono(P)
2   minPx = min(min(P(1,:))) - 1;
3   maxPx = max(max(P(1,:))) + 1;
4
5   minPy = min(min(P(size(P)(1)-1,:))) - 1;
6   maxPy = max(max(P(size(P)(1)-1,:))) + 1;
7
8   figure;
9   axis ([minPx, maxPx, minPy, maxPy]);
10  set (gca, "ygrid", "on");
11  set (gca, "xgrid", "on");
12
13  hold on;
14
15  tam = (size(P)(2));
16  for i = 1: tam
17      tmp = mod(i+1, size(P)(2));
18
19      if tmp != 0
20          prox = tmp;
21      else
22          prox = i+1;
23      endif
24
25      H(i) = line([P(1, i), P(1, prox)] , [P(2, i), P(2, prox)], "linewidth", 2);
26  endfor
```

27

28 `endfunction`

```

1  function retval = rotacionaPoligono(Po, a)
2      minX = min(Po(1,:));
3      maxX = max(Po(1,:));
4      minY = min(Po(2,:));
5      maxY = max(Po(2,:));
6
7      Pc = [minX+(maxX-minX)/2  minY+(maxY-minY)/2];
8
9      T1 = [1 0 -Pc(1); 0 1 -Pc(2); 0 0 1];
10     T2 = [cos(a) -sin(a) 0; sin(a) cos(a) 0; 0 0 1];
11     T3 = [1 0 Pc(1); 0 1 Pc(2); 0 0 1];
12
13     T = T3 * T2 * T1;
14     Pf = T * Po;
15
16     retval = Pf;
17     return;
18 endfunction

```

```

1  #Po = [1 2 2 1; 1 1 2 2; 1 1 1 1]; #quadrado
2  #Po = [2 6 6 2; 2 2 4 4; 1 1 1 1]; #retângulo
3  #Po = [1 2 1; 1 1 2; 1 1 1]; #triângulo
4  Po = [1 2 2 1.5 1; 1 1 2 1.5 2; 1 1 1 1 1]; #bandeirinha
5
6  a = pi/2;
7
8  desenhaPoligono(Po);
9
10 desenhaPoligono(rotacionaPoligono(Po, a));

```

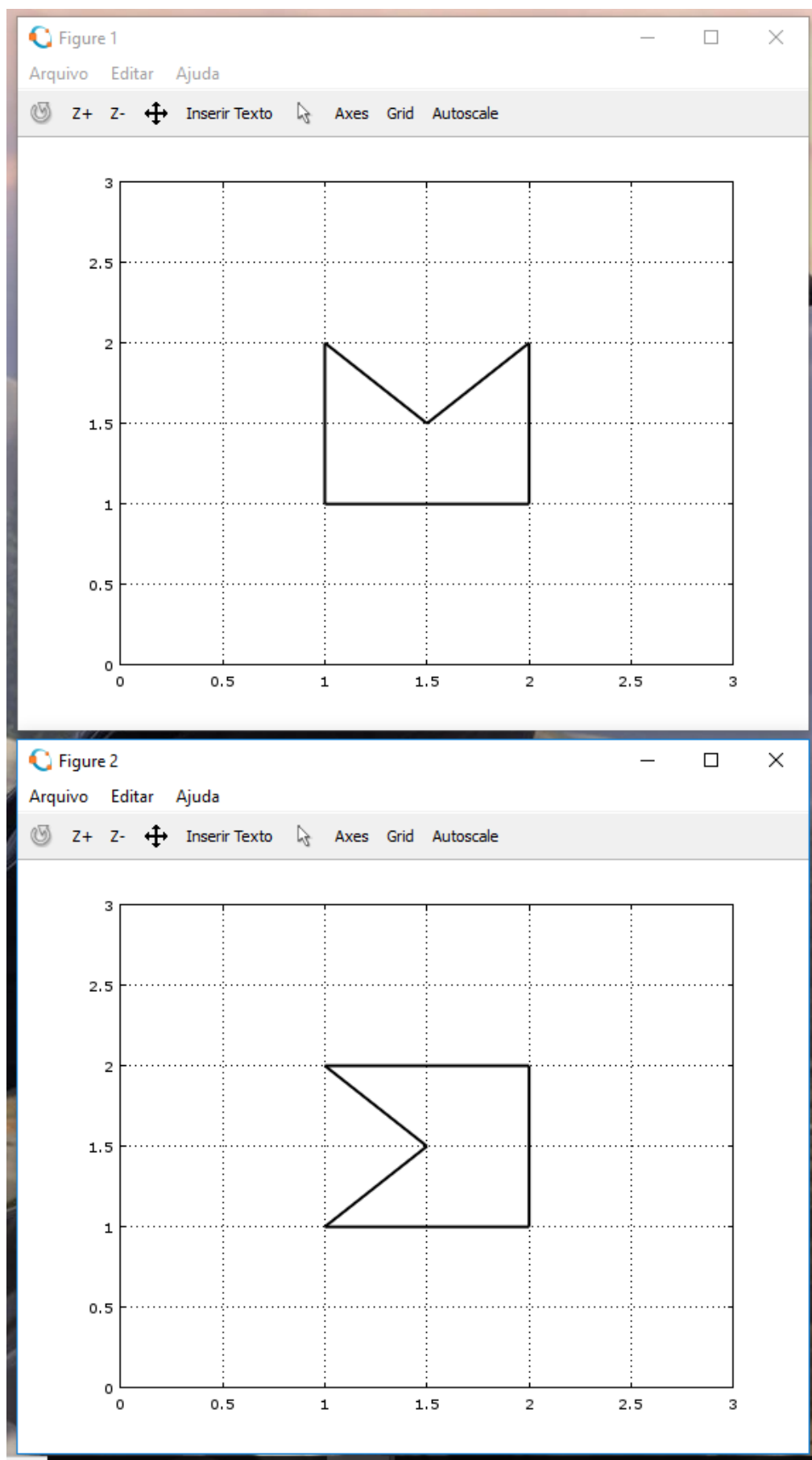


Figura 1: Imagem final

Segundo passo

1. Crie uma função no Octave que realize a rotação do objeto geométrico sobre o seu centro como uma animação, ou seja, variando o ângulo de rotação.
2. Considere utilizar as seguintes funções do Octave: `drawnow` e `pause`.
3. Na plotagem você deve trabalhar com handles dos objetos gráficos (linhas), conforme: `h(i) = line(...)`.
4. A cada iteração (do ângulo de rotação), você deve apagar os objetos gráficos já plotados com a função `delete(h(:))`.
5. Submeta o código da função e um script que chame a função (em arquivos `.m`).

Resposta:

```
1 function animaPoligono(a_inicial, a_final, Po, pause_time, delta_a)
2     minPx = min(min(Po(1,:))) - 1;
3     maxPx = max(max(Po(1,:))) + 1;
4
5     minPy = min(min(Po(size(Po)(1)-1,:))) - 1;
6     maxPy = max(max(Po(size(Po)(1)-1,:))) + 1;
7
8     figure;
9     axis ([minPx, maxPx, minPy, maxPy]);
10    set (gca, "ygrid", "on");
11    set (gca, "xgrid", "on");
12
13    a = a_inicial;
14    tam = (size(Po)(2));
15    H = zeros(size(Po,2));
16
17    while(a < a_final)
18
19        if (a != a_inicial)
20            delete(H(:));
21        endif
22
23        P = rotacionaPoligono(Po, a);
24
25        for i = 1: tam
26
27            tmp = mod(i+1, size(Po)(2));
28
29            if tmp != 0
30                prox = tmp;
31            else
32                prox = i+1;
33            endif
34
35            H(i,:) = line([P(1, i), P(1, prox)] , [P(2, i), P(2, prox)], "linewidth", 2);
36
37        endfor
38        drawnow;
39        a = a + delta_a;
40        pause(pause_time);
41    end
42 endfunction
```

```
1 animaPoligono(0, 5, Po, 0.01, 0.05); #a_inicial, a_final, Po, pause_time, delta_a
```