

Actividad 2.2 (Parametrización) | Parte 1

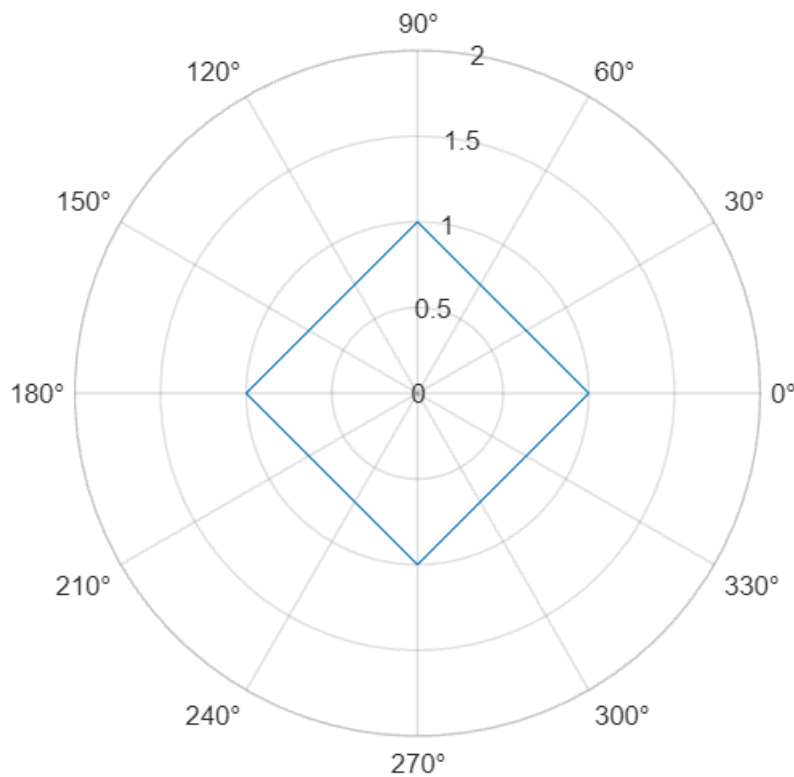
Oscar Ortiz Torres A01769292

```
clear all
close all
clc
```

Para editar el número de lado de los polígonos, el parámetro a editar fue el número de punto en el que se divide el rango de la variable theta, y dicho rango debe ser de $-\pi$ a π para hacer la figura cerrada.

Ejercicio 1

```
n = 5;
% Graficación de la trayectoria en coordenadas polares
% Definición del parámetro theta como un vector en radianes
theta1 = linspace(-pi, pi, n);
% Definición del parámetro r como una función respecto a theta2
r1 = ones(1,n);
% Ingreso de los parámetros a la función polar para la representación
% gráfica (theta(radianes))
figure(1)
polarplot(theta1,r1);
```



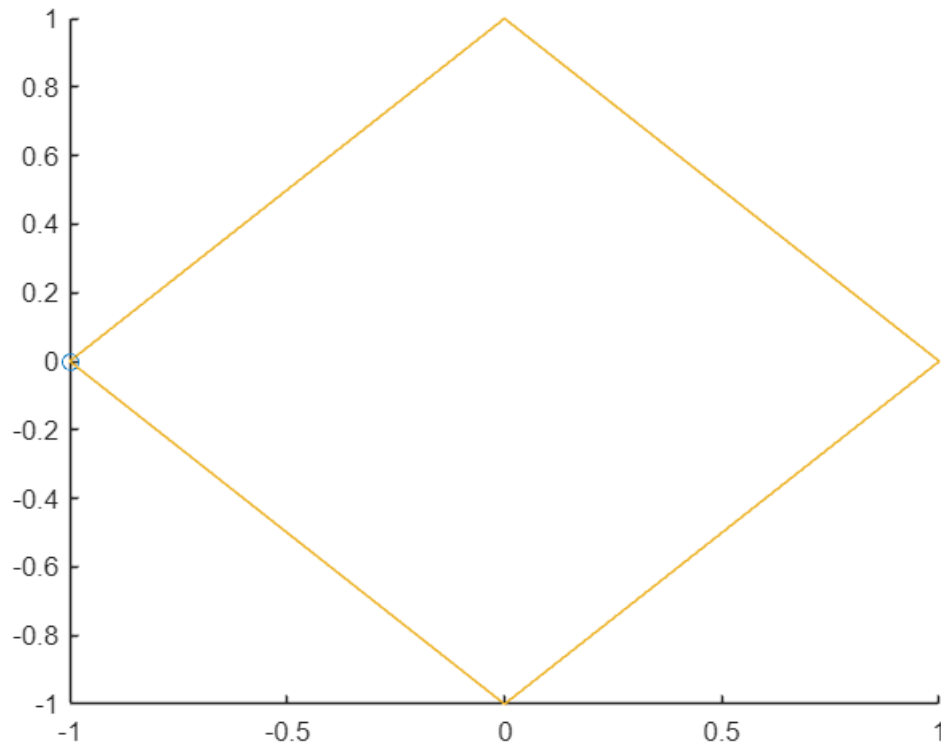
```
% Graficación de la trayectoria en coordenadas cartesianas
% Transformación de las coordenadas polares a cartesianas
```

```

x1 = r1 .* cos(theta1);
y1 = r1 .* sin(theta1);

% Graficación de la trayectoria
figure(2);
comet(x1,y1);

```



```

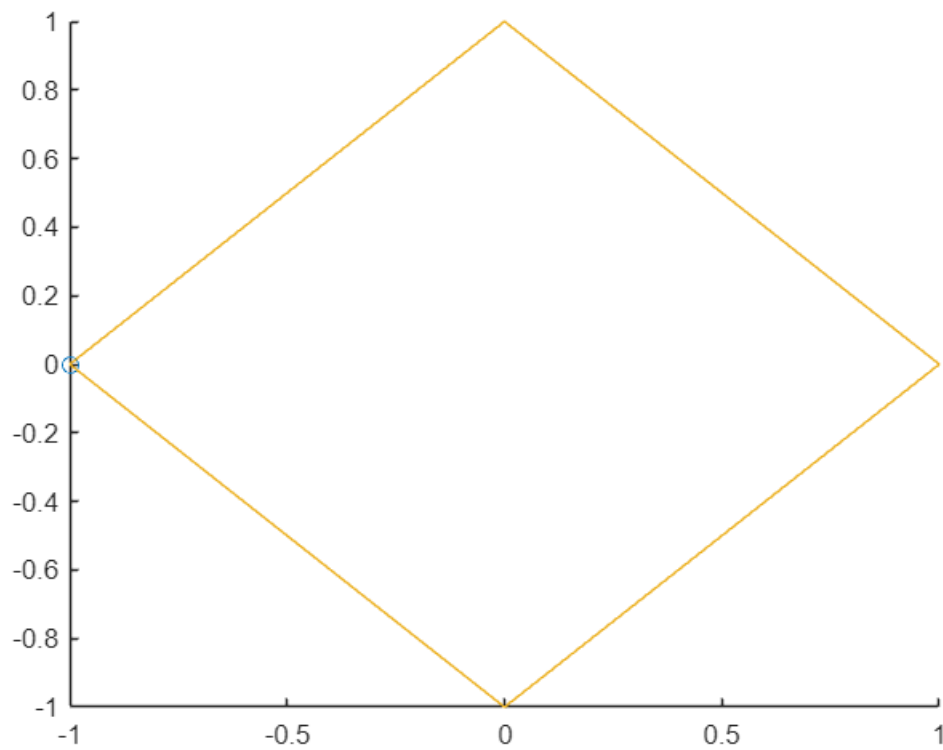
% Definición del parámetro t de parametrización sobre el cual se realizará
% la proyección de trayectoria
tiempo = 1:n;

% Normalización del intervalo de tiempo al intervalo de variación del
% ángulo theta
t1 = normalize(tiempo, "range", [-pi, pi]);

% Obtención de las ecuaciones dependientes del tiempo
x1_1 = ones(1,n) .* cos(t1);
y1_1 = ones(1,n) .* sin(t1);

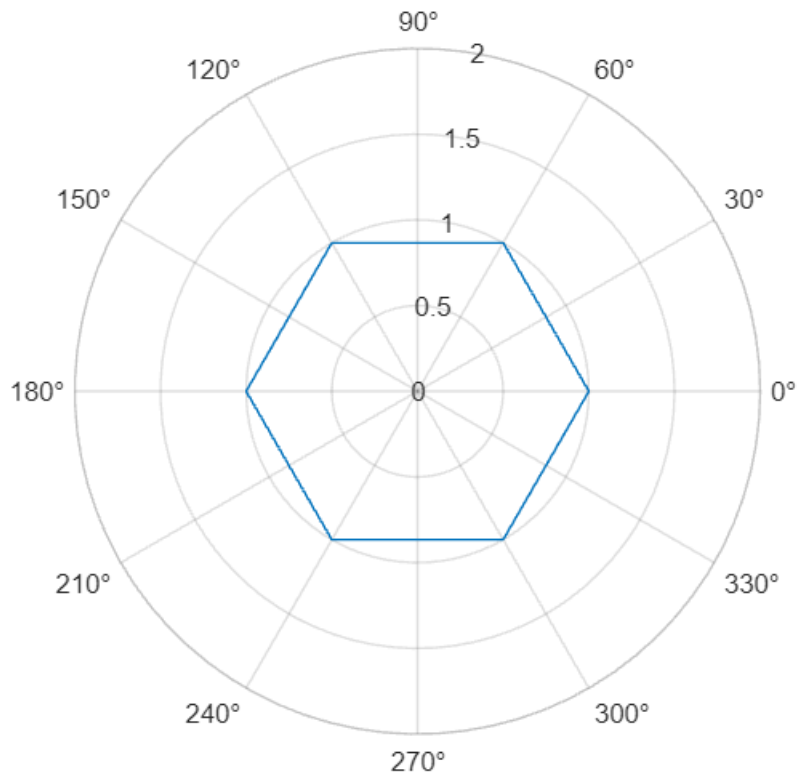
figure(3)
comet(x1_1,y1_1)

```



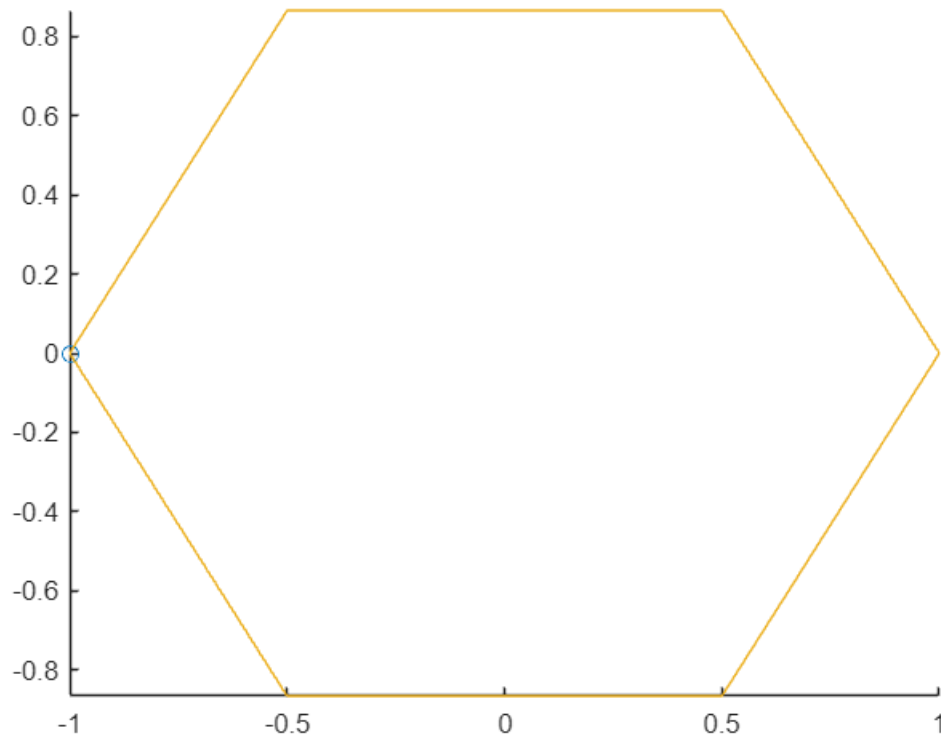
Ejercicio 2

```
n = 7;
% Graficación de la trayectoria en coordenadas polares
% Definición del parámetro theta como un vector en radianes
theta2 = linspace(-pi, pi, n);
% Definición del parámetro r como una función respecto a theta2
r2 = ones(1,n);
% Ingreso de los parámetros a la función polar para la representación
% gráfica (theta(radianes))
figure(4)
polarplot(theta2,r2);
```



```
% Graficación de la trayectoria en coordenadas cartesianas
% Transformación de las coordenadas polares a cartesianas
x2 = r2 .* cos(theta2);
y2 = r2 .* sin(theta2);

% Graficación de la trayectoria
figure(5);
comet(x2,y2);
```

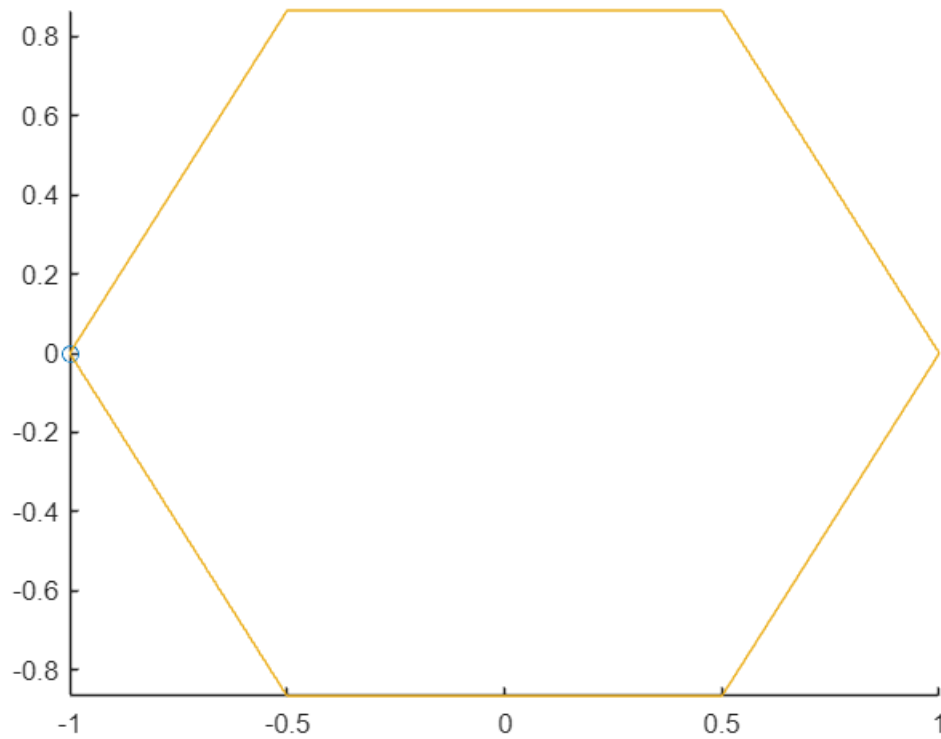


```
% Definición del parámetro t de parametrización sobre el cual se realizará
% la proyección de trayectoria
tiempo = 1:n;
```

```
% Normalización del intervalo de tiempo al intervalo de variación del
% ángulo theta
t2 = normalize(tiempo, "range", [-pi, pi]);
```

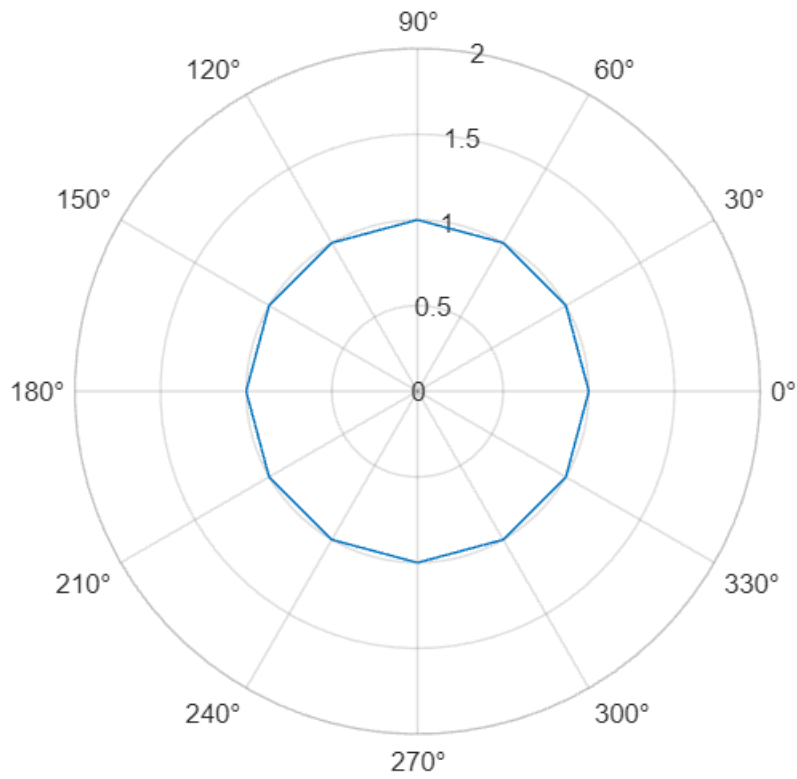
```
% Obtención de las ecuaciones dependientes del tiempo
x2_1 = ones(1,n) .* cos(t2);
y2_1 = ones(1,n) .* sin(t2);
```

```
figure(6)
comet(x2_1,y2_1)
```



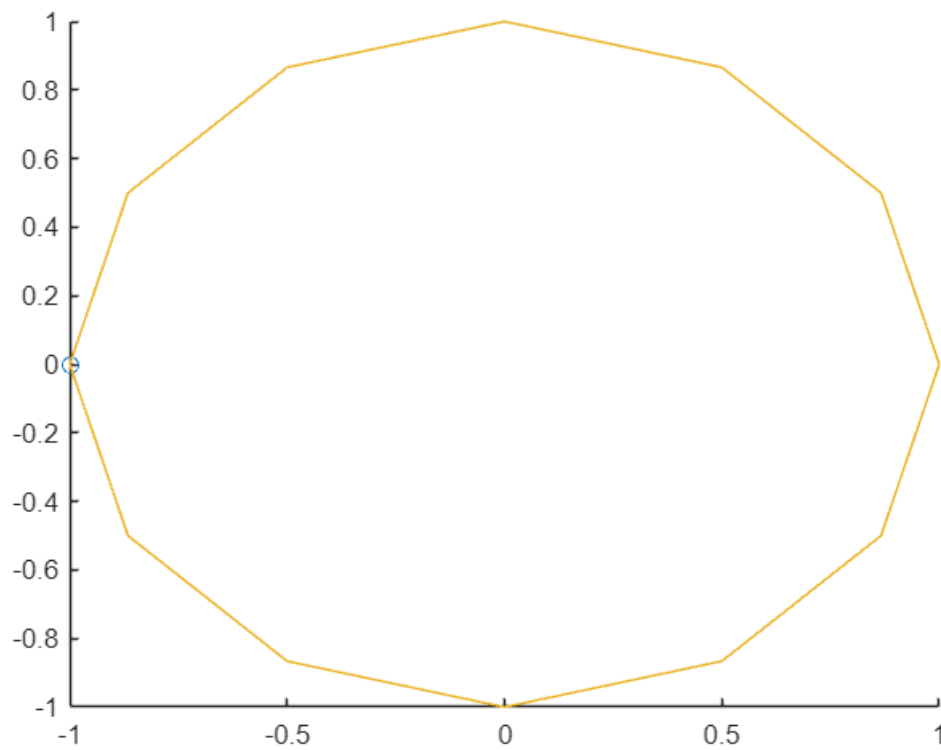
Ejercicio 3

```
n = 13;
% Graficación de la trayectoria en coordenadas polares
% Definición del parámetro theta como un vector en radianes
theta3 = linspace(-pi, pi, n);
% Definición del parámetro r como una función respecto a theta2
r3 = ones(1,n);
% Ingreso de los parámetros a la función polar para la representación
% gráfica (theta(radianes))
figure(7)
polarplot(theta3,r3);
```



```
% Graficación de la trayectoria en coordenadas cartesianas
% Transformación de las coordenadas polares a cartesianas
x3 = r3 .* cos(theta3);
y3 = r3 .* sin(theta3);

% Graficación de la trayectoria
figure(8);
comet(x3,y3);
```



```
% Definición del parámetro t de parametrización sobre el cual se realizará
% la proyección de trayectoria
tiempo = 1:n;

% Normalización del intervalo de tiempo al intervalo de variación del
% ángulo theta
t3 = normalize(tiempo, "range", [-pi, pi]);

% Obtención de las ecuaciones dependientes del tiempo
x3_1 = ones(1,n) .* cos(t3);
y3_1 = ones(1,n) .* sin(t3);

figure(9)
comet(x3_1,y3_1)
```