

```

clear
close all
clc

% Declaración de variables simbólicas
syms l1(t) l2(t) l3(t) t

% Configuración del robot (1 para juntas prismáticas)
RP = [1 1 1];

% Coordenadas articulares
Q = [l1, l2, l3];

% Velocidades generalizadas
Qp = diff(Q, t);

% Número de grados de libertad del robot
GDL = length(RP);

% Matriz identidad 3x3
I3 = eye(3);

% Vector de ceros 1x3 para completar las matrices homogéneas
zeroRow = [0 0 0 1];

% Inicialización de matrices de transformación
T = sym(zeros(4,4,GDL));
A = sym(zeros(4,4,GDL));

% Definición de las transformaciones homogéneas para cada articulación
% Articulación 1: Movimiento en X
P1 = [l1; 0; 0];
A(:, :, 1) = [I3 P1; zeroRow];

% Articulación 2: Movimiento en Y
P2 = [0; l2; 0];
A(:, :, 2) = [I3 P2; zeroRow];

% Articulación 3: Movimiento en Z
P3 = [0; 0; l3];
A(:, :, 3) = [I3 P3; zeroRow];

% Inicialización de la matriz de transformación global
for i = 1:GDL
    if i == 1
        T(:, :, i) = A(:, :, i);
    else
        T(:, :, i) = T(:, :, i-1) * A(:, :, i);
    end
end

```

```

% Mostrar matrices
disp(['Matriz de Transformación Global T', num2str(i)]);
pretty(T(:,:,i));
end

```

```

Matriz de Transformación Global T1
/ 1, 0, 0, 0 \
| 0, 1, 0, 0 |
| 0, 0, 1, l1(t) |
| 0, 0, 0, 1 |
\ 0, 0, 0, 1 /
Matriz de Transformación Global T2
/ 1, 0, 0, 0 \
| 0, 1, 0, 0 |
| 0, 0, 1, l1(t) + l2(t) |
| 0, 0, 0, 1 |
\ 0, 0, 0, 1 /
Matriz de Transformación Global T3
/ 1, 0, 0, 0 \
| 0, 1, 0, 0 |
| 0, 0, 1, l1(t) + l2(t) + l3(t) |
| 0, 0, 0, 1 |
\ 0, 0, 0, 1 /

```

```

% Inicializamos el Jacobiano
Jv_a = sym(zeros(3, GDL));
Jw_a = sym(zeros(3, GDL));

for k = 1:GDL
    if k == 1
        Jv_a(:,k) = [1; 0; 0]; % Movimiento en X
    elseif k == 2
        Jv_a(:,k) = [0; 1; 0]; % Movimiento en Y
    elseif k == 3
        Jv_a(:,k) = [0; 0; 1]; % Movimiento en Z
    end

    Jw_a(:,k) = [0; 0; 0];
end

Jv_a = simplify(Jv_a);
Jw_a = simplify(Jw_a);

% Velocidades lineales y angulares
V = simplify(Jv_a * Qp.');
W = simplify(Jw_a * Qp.');

```

