```
%                    问题一：模拟舞龙队盘入的过程                         %
%  (运行前请clear工作区,避免出现错误!)

% 参数设置
benches_num = 224;  % 板凳数量
head_long = 3.41; % 龙头长度
body_long = 2.20; % 龙身和龙尾长度
benches_width = 0.3; % 板宽
hole_to_head = 0.275; % 板凳孔距离最近板凳头距离
p = 0.55;           % 螺距
v_head = 1.0;       % 龙头速度
T = 300;            % 模拟时间
dt = 0.01;          % 时间步长

% 第一个把手的初始位置和角度设置
theta0 = 2*pi*16;
r0 = p*16;
% 初始化龙头的位置
positions(1, :, 1) = [r0 * cos(theta0), r0 * sin(theta0)];
% 计算每个板凳孔相对于上一节的偏移
L = [head_long - 2 * hole_to_head; ...
    repmat(body_long - 2 * hole_to_head, benches_num-1, 1)];  % 每节的长度

% 初始化龙身和龙尾位置
initial_theta = theta0;
initial_r = r0;

%计算龙身的初始位置
for i = 2:benches_num
    delta_theta(i) = L(i-1) / initial_r;  % 每节之间的角度差 弧长=半径×角度
    initial_theta = initial_theta + delta_theta(i);
    initial_r = p / (2 * pi) * initial_theta;     % 半径变化
    positions(i, 1, 1) = initial_r * cos(initial_theta);  % x位置
    positions(i, 2, 1) = initial_r * sin(initial_theta);  % y位置
end

%初始化速度矩阵
velocities = zeros(224, 301);

%初始化角度和半径
current_theta = theta0;
current_r = r0;

%每一个dt时刻的龙位置
for j = dt:dt:T
    t = round(j / dt);
    % 计算龙头位置
    theta_head = current_theta - v_head * dt / current_r;
    r_head = p / (2 * pi) * theta_head;
    positions(1, :, t+1) = [r_head * cos(theta_head), ...
                            r_head * sin(theta_head)];
    % 更新龙头前把手极坐标的角度和半径
    current_theta = theta_head;
    current_r = r_head;
    % 更新第一节龙身前把手的角度和半径
    initial_theta = current_theta;
    initial_r = current_r;
    % 计算当前时刻龙身和龙尾位置
    for i = 2:benches_num
        delta_theta(i) = L(i-1) / initial_r;  % 近似：角度=弧长/半径
        initial_theta = initial_theta + delta_theta(i);
        initial_r = p / (2 * pi) * initial_theta;
        positions(i, 1, t+1) = initial_r * cos(initial_theta);  % x位置
        positions(i, 2, t+1) = initial_r * sin(initial_theta);  % y位置
    end

    % 计算每节的速度
    if t > 0
        a = (positions(:, 1, t+1) - positions(:, 1, t)) / dt;  % vx
        b = (positions(:, 2, t+1) - positions(:, 2, t)) / dt;  % vy
        velocities(:, t+1) = sqrt(a.^2 + b.^2);
    end


    % %图像显示占用电脑资源影响运行速度，可以选择注释提高运行效率
    % % 绘制当前时刻龙的位置
    % pause(0.01);
    % clf;
    % hold on;
    % axis equal;
    % xlabel('X (米)');
    % ylabel('Y (米)');
    % % 设置坐标轴范围
    % xlim([-12, 12]);
    % ylim([-12, 12]);
    % title(['板凳龙行进示意图 (t = ', num2str(j), 's)']);
    % grid on;
    % % 画背景螺线图
    % theta_spiral = linspace(0, -32*pi, 10000);
    % r_spiral = 0.55 * 16 + (0.55 / (2 * pi)) * theta_spiral;
    % x_spiral = r_spiral .* cos(theta_spiral);
    % y_spiral = r_spiral .* sin(theta_spiral);
    % plot(x_spiral, y_spiral,'LineWidth', 0.5, 'Color', 'm');
    % %画龙
    % plot(positions(1, 1, t+1), positions(1, 2, t+1), 'ro-', ...
    %     'MarkerSize', 4, 'LineWidth', 2,'MarkerFaceColor', 'r');
    % plot(positions(2:end, 1, t+1), positions(2:end, 2, t+1),...
    %     'co-', 'MarkerSize', 4, 'LineWidth', 2,'MarkerFaceColor', 'b');
    % line([positions(1, 1, t+1), positions(2, 1, t+1)],...
    %     [positions(1, 2, t+1), positions(2, 2, t+1)],...
    %     'Color','red','LineWidth', 2,'LineStyle','-');
    % hold off;


end

% 输出0s - 300s数据
output_times = 0:1:300;
positions_output = zeros(benches_num, 2, length(output_times));
velocities_output = zeros(benches_num,length(output_times));

%存入数据
for i = 1:length(output_times)
    t_idx = round(output_times(i)/dt) + 1;
    positions_output(:, :, i) = positions(:, :, t_idx);
    velocities_output(:, i) = velocities(:, t_idx);
end
velocities_output(:, 1)=1;

% 保存结果到Excel文件
filename = 'result1.xlsx';
location=[];
for i = 1:length(output_times)
    location = [location,reshape(squeeze(positions_output(:, :, i))'...
                ,1,448)'];
end

writematrix(location, filename, 'Sheet','位置', 'Range', 'B2');
writematrix(velocities_output, filename, 'Sheet','速度', 'Range', 'B2');
disp('数据已存入result1.xlsx');
```