

```

function test()
    if(startSimulation(600,400)~=0) return; endif % シミュレーション開始(ウィンドウサイズ)
    %if(startSimulation(0,0)~=0) return; endif % シミュレーション開始(ウィンドウを表示しない)
    more off
    printf('press any key.¥n')
    kbhit();
    printf('press a,s,d,f,g: change angles¥n')
    printf('press w,b: show state¥n')
    printf('press r: reset¥n')
    printf('press q: quit¥n')
    kp=80; kd=2;
    MaxTorque = 100.0;
    q1=0.25*pi; q2=-0.5*pi;
    q3=0.125*pi; q4=-0.25*pi;
    target = [q3,q4,q3,q4,q3,q4,q3,q4]';
    ii=0;
    while (1)
        jstate = getJointState(); % robot の関節状態 [q1,...,q8,dq1,...,dq8]' を返す
        u = zeros(8,1);
        for i=1:8
            u(i)=kp*(target(i)-jstate(i))-kd*jstate(8+i);
            if (u(i) > MaxTorque) u(i) = MaxTorque; endif
            if (u(i) < -MaxTorque) u(i) = -MaxTorque; endif
        endfor
        stepSimulation(u,0.0005); % (トルク, 時間幅)でシミュレーションを進める
        if (ii==0) drawWorld(); endif % 25 回に 1 回の割合で描画 (描画が重いことを想定して)
        ii++;
        if (ii==25) ii=0; endif
        key=kbhit(1);
        switch key
            case 'q'; stopSimulation(); printf('¥n'); return; % シミュレーションを終了
            case 'a'; target = [q1,q2,q1,q2,q1,q2,q1,q2]';
            case 's'; target = [q3,q4,q3,q4,q3,q4,q3,q4]';
            case 'd'; target = zeros(8,1);
            case 'f'; target = -[q3,q4,q3,q4,q3,q4,q3,q4]';
            case 'g'; target = -[q1,q2,q1,q2,q1,q2,q1,q2]';
            case 'w'; getJointState()
            case 'b'; getBaseState()
            case 'r'; resetSimulation();
        endswitch
    endwhile
end

```