```
%%% Layer Jamming Design Methodology %%%
%%% %%%

%% Inputs definition

%step 1:Define differential pressure
prompt_p = {'Enter differential pressure [MPa]:'};
dlg_title='input';
num_lines=1;
def={''};
input_p=inputdlg(prompt_p,dlg_title,num_lines,def); %[bar]
P=str2num(input_p{1});



% step 2: Main load case selection
load_case = menu('Load case selection','Axial
forces','Transversal Forces','Bending moments','Torsional
moments');


% step 3: degrees of freedom selection
if load_case ==1 || load_case==2
gdl = menu('Configuration degrees of freedom','Bulk
system','Useful Area variation');
else
end


% step 4: Design selection (by size or by material)
design_sel = menu('Design selection','By size','By material');


%% Design definition
if load_case==1
    %% axial forces
    if design_sel==1
```

```
        %design by size
        prompt_dim = {'Enter lenght l [mm]:','Enter width w
[mm]:','Enter maximum force F_l [N]:','Enter minimum force F_l
[N]:'};
        dlg_title='input';
        num_lines=1;
        def={'','','',''};

input_dim=inputdlg(prompt_dim,dlg_title,num_lines,def);
        l=str2num(input_dim{1}); %[mm]
        w=str2num(input_dim{2}); %[mm]
        F_max_l=str2num(input_dim{3}); %[N]
        F_min_l=str2num(input_dim{4}); %[N]



        out2=0;
        while out2==0
             out1=0;
            while out1==0
            prompt_h = {'Enter overall height h [mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_h=inputdlg(prompt_h,dlg_title,num_lines,def);
            h=str2num(input_h{1}); %[mm]


            t_mu_ratio_min=(h*l*w*P)/F_max_l;
            t_mu_ratio_max=(h*l*w*P)/F_min_l;
            fprintf('t/mu =[%f;%f] [mm] \n',
t_mu_ratio_min,t_mu_ratio_max);


            out_sel1 = menu('Result of the iteration','Ok','No
match');
```

```
                if out_sel1==1
                    out1=1;
                else
                    out1=0;
                end
            end
        prompt_res = {'Enter selected frictional coefficent
[]:','Enter selected layer height t [mm]:'};
        dlg_title='input';
        num_lines=1;
        def={'',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        mu_res=str2num(input_res{1}); % adim
        t_res=str2num(input_res{2}); %[mm]
        F_eff=h/t_res*l*w*mu_res*P;
        fprintf('F = %f N \n', F_eff);
            if gdl==1
            out2=1;
            else
            prompt_L = {'Enter maximum lenght L [mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_L=inputdlg(prompt_L,dlg_title,num_lines,def);
            l_min=2*l-(str2num(input_L{1})); % [mm]
            F_L=h/t_res*mu_res*l_min*w*P;
            fprintf('F_L = %f N \n', F_L);
            out_sel2 = menu('Result of the iteration','Ok','No
match');
                if out_sel2==1
                out2=1;
                else
```

```
                end
            end
        end
        n=ceil(h/(t_res));
        t=t_res;
        fprintf('n = %f \n', n);
        fprintf('t = %f mm \n', t);
        h=n*t;
        mu=mu_res;
    else
        %design by material
        prompt_mat = {'Enter frictional coefficient
[]:','Enter layer height t [mm]:','Enter maximum force F_l
[N]:','Enter minimum force F_l [N]','Enter l/w ratio []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','',''};

input_material=inputdlg(prompt_mat,dlg_title,num_lines,def);
        mu=str2num(input_material{1}); %adim
        t=str2num(input_material{2}); %[mm]
        F_max_l=str2num(input_material{3}); %[N]
        F_min_l=str2num(input_material{4}); %[N]
        l_w_ratio=str2num(input_material{5}); %adim

        out2=0;
        while out2==0
        hww_max=F_max_l*t/(mu*P)*l_w_ratio;  %[mm^3]
        hww_min=F_min_l*t/(mu*P)*l_w_ratio;  %[mm^3]
        fprintf('h*w^2 = [%f;%f] [mm^3] \n', hww_min,hww_max);
        prompt_res = {'Enter selected w [mm]:','Enter selected
h [mm]:'};
        dlg_title='input';
        num_lines=1;
```

```
        def={'','','};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        w_res=str2num(input_res{1}); % [mm]
        h_res=str2num(input_res{2}); %[mm]
        n=ceil(h_res/t);
        w=w_res;
        l=w_res*l_w_ratio;
        fprintf('n = %f\n', n);
        fprintf('w = %f mm \n',w);
        fprintf('l = %f mm \n',l);
            if gdl==1
            out2=1;
            else
            prompt_F_L = {'Enter force at lenght L F_L [N]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_F_L=inputdlg(prompt_F_L,dlg_title,num_lines,def);
            F_L=str2num(input_F_L{1}); % [N]
            l_min=F_L*t/(mu*h_res*w_res*P); %[mm]
            L=2*l-l_min;
            fprintf('L = %f mm \n',L);
            out_sel2 = menu('Result of the iteration','Ok','No
match');
                if out_sel2==1
                out2=1;
                else
                end
            end
        end
        h=h_res;
```

```
    end

sel3 = menu('Evaluate other load cases
performances?','Yes','No');
        if sel3==1
            prompt_E = {'Enter Young modulus E[MPa]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_E=inputdlg(prompt_E,dlg_title,num_lines,def);
            E=str2num(input_E{1}); %[MPa]
            %Bending moment
            Kb=E*w*h^3/12; % [N*mm^2]
            fprintf('Kb = %f N x mm^2 \n',Kb);
            %Transversal forces
            K=E*w*h^3/(4*l^3);
            fprintf('K = %f N/mm \n',K);
            Ft=3*P*mu*w*h/2;
            fprintf('Ft = %f N \n',Ft);
            %Torsional moment
            prompt_ni = {'Enter poisson ratio []:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_ni=inputdlg(prompt_ni,dlg_title,num_lines,def);
            ni=str2num(input_ni{1}); %adim
            G=E/(2*1+ni); %[MPa]
            Kt=G*h^3*w/3; % [N*mm^2]
            fprintf('Kt = %f N x mm^2 \n',Kt);
            Ms=mu*P*w^2*h/3; %[N*mm]
            fprintf('Ms = %f N x mm \n',Ms)
```

```
        else
        end


elseif load_case==2
    %% transversal forces
    if design_sel==1
        %design by size
        prompt_dim = {'Enter lenght l [mm]:','Enter width w
[mm]:','Enter maximum slip force Fs [N]:','Enter minimum slip
force Fs [N]:','Enter maximum stiffness K [N/mm]:','Enter
minimum stiffness K [N/mm]:','Enter maximum jamming ratio r
[]:','Enter minimum jamming ratio r []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','','','','','',''};

input_dim=inputdlg(prompt_dim,dlg_title,num_lines,def);
        l=str2num(input_dim{1}); %[mm]
        w=str2num(input_dim{2}); %[mm]
        Fs_max_l=str2num(input_dim{3}); %[N]
        Fs_min_l=str2num(input_dim{4}); %[N]
        K_max_l=str2num(input_dim{5}); %[N/mm]
        K_min_l=str2num(input_dim{6}); %[N/mm]
        n_max=ceil(sqrt(str2num(input_dim{7})));
        n_min=ceil(sqrt(str2num(input_dim{8})));
        fprintf('n = [%f;%f] \n',n_min,n_max);
        out2=0;
        while out2==0
             out1=0;
            while out1==0
            prompt_h = {'Enter overall height h [mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};
```

```
input_h=inputdlg(prompt_h,dlg_title,num_lines,def);
            h=str2num(input_h{1}); %[mm]
            t_max=(h/n_min);
            t_min=(h/n_max);
            fprintf('t = [%f;%f] [mm] \n',t_min,t_max);
            mu_max=3*Fs_max_l/(2*h*w*P); %mu adim
            mu_min=3*Fs_min_l/(2*h*w*P); %mu adim
            fprintf('mu = [%f;%f] \n',mu_min,mu_max);
            E_max=4*K_max_l*l^3/(w*h^3); %E [MPa]
            E_min=4*K_min_l*l^3/(w*h^3); %E [MPa]
            fprintf('E = [%f;%f] [MPa] \n',E_min,E_max);
            out_sel1 = menu('Result of the iteration','Ok','No
match');
                if out_sel1==1
                    out1=1;
                else
                    out1=0;
                end
            end
        prompt_res = {'Enter selected frictional coefficent
[]:','Enter selected Young modulus E [MPa]:','Enter selected
layer height t [mm]:','Enter selected number of layer n []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        mu=str2num(input_res{1}); % adim
        E_res=str2num(input_res{2}); %[MPa]
        t_res=str2num(input_res{3}); %[mm]
        n_res=str2num(input_res{4}); % adim
        E=E_res;
        t=t_res;
```

```
        n=n_res;
        h=n_res*t;

            if gdl==1
            out2=1;
            else
            prompt_L = {'Enter maximum lenght L [mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_L=inputdlg(prompt_L,dlg_title,num_lines,def);
            L=str2num(input_L{1}); % [mm]
            K_L=E_res*w*h^3/(4*L^3); %N/mm
            fprintf('K_L = %f N/mm \n',K_L);
            out_sel2 = menu('Result of the iteration','Ok','No
match');
                if out_sel2==1
                out2=1;
                else
                end
            end
        end
    else
        %design by material
        prompt_mat = {'Enter frictional coefficient
[]:','Enter layer height t [mm]:','Enter Young modulus E
[MPa]','Enter maximum slip force Fs [N]:','Enter minimum slip
force Fs [N]:','Enter maximum stiffness K [N/mm]:','Enter
minimum stiffness K [N/mm]:','Enter maximum jamming ratio r
[]:','Enter minimum jamming ratio r []:','Enter l/w ratio
[]:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','','','','','','',''};
```

```
input_material=inputdlg(prompt_mat,dlg_title,num_lines,def);
        mu=str2num(input_material{1}); %adim
        t=str2num(input_material{2}); %[mm]
        E=str2num(input_material{3}); %[MPa]
        F_max_l=str2num(input_material{4}); %[N]
        F_min_l=str2num(input_material{5}); %[N]
        K_max_l=str2num(input_material{6}); %[N/mm]
        K_min_l=str2num(input_material{7}); %[N/mm]
        n_max=ceil(sqrt(str2num(input_material{8})));
        n_min=ceil(sqrt(str2num(input_material{9})));
        l_w_ratio=str2num(input_material{10}); %adim
        fprintf('n = [%f;%f] \n',n_min,n_max);
        out2=0;
        while out2==0

        nw_max=3*F_max_l/(2*t*mu*P); %ny [mm]
        nw_min=3*F_min_l/(2*t*mu*P); %ny [mm]
        fprintf('n*w = [%f;%f] [mm] \n',nw_min,nw_max);
        nnn_ww_ratio_max=4*K_max_l*l_w_ratio^3/(t^3*E);
%n^3/y^2 [1/mm^2]
        nnn_ww_ratio_min=4*K_min_l*l_w_ratio^3/(t^3*E);
%n^3/y^2 [1/mm^2]
        fprintf('n^3/w^2 = [%f;%f] [1/mm^2]
\n',nnn_ww_ratio_min,nnn_ww_ratio_max);

        prompt_res = {'Enter selected w [mm]:','Enter selected
number of layers n []:'};
        dlg_title='input';
        num_lines=1;
        def={'',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        w_res=str2num(input_res{1}); % [mm]
```

```
        n_res=str2num(input_res{2}); % adim
        h_res=n_res*t; % [mm]
        w=w_res;
        l=w_res*l_w_ratio;
            if gdl==1
            out2=1;
            else
            prompt_K_L = {'Enter minimum stiffness at lenght L
K_L [N/mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_K_L=inputdlg(prompt_K_L,dlg_title,num_lines,def);
            K_L=str2num(input_K_L{1}); % [N/mm]
            L=(E*w_res*h_res^3/(4*K_L))^(1/3); %[mm]
            fprintf('L = %f mm \n',L);
            l_min=2*l-L;
            out_sel2 = menu('Result of the iteration','Ok','No
match');
                if out_sel2==1
                out2=1;
                else
                end
            end
        end
        w=w_res;
        l=w_res*l_w_ratio;
        fprintf('w = %f mm \n',w);
        fprintf('l = %f mm \n',l);
        n=n_res;
        h=h_res;
```

```
    end

    K_eff=E*w*h^3/(4*l^3);
    r_eff=n^2;
    K_eff_0=K_eff/r_eff;
    fprintf('K = %f N/mm \n',K_eff);
    fprintf('K_0 = %f N/mm \n',K_eff_0);
    fprintf('r = %f \n',r_eff);
    if gdl==1
    else
    K_eff_L=E*w*h^3/(4*L^3);
    K_eff_L_0=K_eff_L/r_eff;
    fprintf('K_L = %f N/mm \n',K_eff_L);
    fprintf('K_L_0 = %f N/mm \n',K_eff_L_0);
    end
    Fs_eff=2/3*P*w*h*mu;
    fprintf('Fs = %f N \n',Fs_eff);

    %other load cases performances

    sel3 = menu('Evaluate other load cases
performances?','Yes','No');
        if sel3==1
            %Axial forces
            F_l=n*mu*P*l*w; % [N]
            fprintf('F_l = %f N \n',F_l);
            %Bending moment
            Kb=E*w*h^3/12; % [N*mm^2]
            fprintf('Kb = %f N x mm^2 \n',Kb);
            %Torsional moment
            prompt_ni = {'Enter poisson ratio []:'};
            dlg_title='input';
            num_lines=1;
```

```
            def={''};

input_ni=inputdlg(prompt_ni,dlg_title,num_lines,def);
            ni=str2num(input_ni{1}); %adim
            G=E/(2*(1+ni)); %[MPa]
            Kt=G*h^3*w/3; % [N*mm^2]
            fprintf('Kt = %f N x mm^2 \n',Kt);
            Ms=mu*P*w^2*h/3; %[N*mm]
            fprintf('Ms = %f N x mm \n',Ms);
        else
        end


elseif load_case==3
    %% bending moment
    if design_sel==1
        %design by size
        prompt_dim = {'Enter width w [mm]:','Enter maximum
bending stiffness K_b [N x mm^2]:','Enter minimum bending
stiffness K_b [N x mm^2]:','Enter maximum jamming ratio r
[]:','Enter minimum jamming ratio r []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','',''};

input_dim=inputdlg(prompt_dim,dlg_title,num_lines,def);
        w=str2num(input_dim{1}); %[mm]
        K_max_l=str2num(input_dim{2}); %[N*mm^2]
        K_min_l=str2num(input_dim{3}); %[N*mm^2]
        n_max=ceil(sqrt(str2num(input_dim{4})));
        n_min=ceil(sqrt(str2num(input_dim{5})));
        fprintf('n = [%f;%f] \n',n_min,n_max);
            out1=0;
            while out1==0
            prompt_h = {'Enter overall height h [mm]:'};
```

```
            dlg_title='input';
            num_lines=1;
            def={''};

input_h=inputdlg(prompt_h,dlg_title,num_lines,def);
            h=str2num(input_h{1}); %[mm]


            E_min=K_min_l*12/(h^3*w); %E [MPa]
            E_max=K_max_l*12/(h^3*w); %E [MPa]
            fprintf('E = [%f;%f] [MPa]\n',E_min,E_max);
            out_sel1 = menu('Result of the iteration','Ok','No
match');
                if out_sel1==1
                    out1=1;
                else
                    out1=0;
                end
            end
        prompt_res = {'Enter selected Young Modulus
E[MPa]:','Enter selected number of layers n[]:','Enter
selected layer height t [mm]:'};
        dlg_title='input';
        num_lines=1;
        def={'','',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        E_res=str2num(input_res{1}); % [MPa]
        n_res=str2num(input_res{2}); %adim
        t_res=str2num(input_res{3}); %[mm]
        t=t_res;
        n=n_res;
        h=n_res*t_res;
        E=E_res;
    else
```

```
        %design by material
        prompt_mat = {'Enter layer height t [mm]:','Enter
Young modulus E[MPa]','Enter maximum bending stiffness K_b [N
x mm^2]:','Enter minimum bending stiffness K_b [N x
mm^2]:','Enter maximum jamming ratio r []:','Enter minimum
jamming ratio r []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','','',''};

input_material=inputdlg(prompt_mat,dlg_title,num_lines,def);
        t=str2num(input_material{1}); %[mm]
        E=str2num(input_material{2}); %[MPa]
        K_max_l=str2num(input_material{3}); %[N*mm^2]
        K_min_l=str2num(input_material{4}); %[N*mm^2]
        n_max=ceil(sqrt(str2num(input_material{5})));
        n_min=ceil(sqrt(str2num(input_material{6})));
        fprintf('n = [%f;%f] \n',n_min,n_max);
        wnnn_max=K_max_l*12/(E*t^3); % w*n^3 [mm]
        wnnn_min=K_min_l*12/(E*t^3); % w*n^3 [mm]
        fprintf('w*n^3 = [%f;%f] [mm]\n',wnnn_min,wnnn_max);
        out_sel1 = menu('Result of the iteration','Ok','No
match');

        prompt_res = {'Enter selected w [mm]:','Enter selected
number of layers n []'};
        dlg_title='input';
        num_lines=1;
        def={'',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        w_res=str2num(input_res{1}); % [mm]
        n_res=str2num(input_res{2}); %adim
        w=w_res;
        n=n_res;
```

```
        h=n_res*t;
    end
    Kb_eff=E*w*h^3/12;
    r_eff=n^2;
    Kb_eff_0=Kb_eff/r_eff;
    fprintf('Kb = %f N/mm \n',Kb_eff);
    fprintf('Kb_0 = %f N/mm \n',Kb_eff_0);
    fprintf('r = %f \n',r_eff);

    sel3 = menu('Evaluate other load cases
performances?','Yes','No');
    if sel3==1
            prompt_E = {'Enter lenght l [mm]:','Enter
frictional coefficient []:'};
            dlg_title='input';
            num_lines=1;
            def={'',''};

input_E=inputdlg(prompt_E,dlg_title,num_lines,def);
             l=str2num(input_E{1}); %[mm]
             mu=str2num(input_E{2}); %[mm]

            %Axial forces
            F_l=n*mu*P*l*w; % [N]
            fprintf('F_l = %f N \n',F_l);
            %Transversal forces
            K=E*w*h^3/(4*l^3);
            fprintf('K = %f N/mm \n',K);
            Ft=3*P*mu*w*h/2;
            fprintf('Ft = %f N \n',Ft);
            %Torsional moment
            prompt_ni = {'Enter poisson ratio []:'};
            dlg_title='input';
```

```
            num_lines=1;
            def={''};

input_ni=inputdlg(prompt_ni,dlg_title,num_lines,def);
            ni=str2num(input_ni{1}); %adim
            G=E/(2*1+ni); %[MPa]
            Kt=G*h^3*w/3; % [N*mm^2]
            fprintf('Kt = %f N x mm^2 \n',Kt);
            Ms=mu*P*w^2*h/3; %[N*mm]
            fprintf('Ms = %f N x mm \n',Ms);
        else
        end


elseif load_case==4
    %% torsional moment
    if design_sel==1
        %design by size
        prompt_dim = {'Enter width w [mm]:','Enter maximum
slip moment Ms [N x mm]:','Enter minimum slip moment Ms [N x
mm]:','Enter maximum torsional stiffness Kt [N x
mm^2]:','Enter minimum torsional stiffness Kt [N x
mm^2]:','Enter maximum jamming ratio r []:','Enter minimum
jamming ratio r []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','','','',''};

input_dim=inputdlg(prompt_dim,dlg_title,num_lines,def);
        w=str2num(input_dim{1}); %[mm]
        Ms_max_xmax=str2num(input_dim{2}); %[N*mm]
        Ms_min_xmax=str2num(input_dim{3}); %[N*mm]
        Kt_max_xmax=str2num(input_dim{4}); %[N*mm^2]
        Kt_min_xmax=str2num(input_dim{5}); %[N*mm^2]
        n_max=ceil(sqrt(str2num(input_dim{6})));
        n_min=ceil(sqrt(str2num(input_dim{7})));
```

```
        fprintf('n = [%f;%f] \n',n_min,n_max);


            out1=0;
            while out1==0
            prompt_h = {'Enter overall height h [mm]:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_h=inputdlg(prompt_h,dlg_title,num_lines,def);
            h=str2num(input_h{1}); %[mm]
            t_max=ceil(h/n_min);
            t_min=ceil(h/n_max);
            fprintf('t = [%f;%f] [mm] \n',t_min,t_max);


            mu_max=3*Ms_max_xmax/(h*w^2*P); %mu adim
            mu_min=3*Ms_min_xmax/(h*w^2*P); %mu adim
            fprintf('mu = [%f;%f] \n',mu_min,mu_max);
            G_max=3*Kt_max_xmax/(w*h^3); %E [MPa]
            G_min=3*Kt_min_xmax/(w*h^3); %E [MPa]
            fprintf('G = [%f;%f] [MPa]\n',G_min,G_max);
            out_sel1 = menu('Result of the iteration','Ok','No
match');
                if out_sel1==1
                    out1=1;
                else
                    out1=0;
                end
            end
        prompt_res = {'Enter selected frictional coefficent
[]:','Enter selected Shear modulus G [MPa]:','Enter selected
layer height t [mm]:','Enter selected number of layer n[]:'};
        dlg_title='input';
        num_lines=1;
```

```
        def={'','','','',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        mu=str2num(input_res{1}); % adim
        G_res=str2num(input_res{2}); %[MPa]
        t_res=str2num(input_res{3}); %[mm]
        n_res=str2num(input_res{4}); % adim
        n=n_res;
        t=t_res;
        G=G_res;
    else
        %design by material
        prompt_mat = {'Enter frictional coefficient
[]:','Enter layer height t [mm]:','Enter Shear modulus
G[MPa]','Enter maximum slip moment Ms [N x mm]:','Enter
minimum slip moment Ms [N x mm]:','Enter maximum torsional
stiffness Kt [N x mm^2]:','Enter minimum torsional stiffness
Kt [N x mm^2]:','Enter maximum jamming ratio r []:','Enter
minimum jamming ratio r []:'};
        dlg_title='input';
        num_lines=1;
        def={'','','','','','','','','',''};

input_material=inputdlg(prompt_mat,dlg_title,num_lines,def);
        mu=str2num(input_material{1}); %adim
        t=str2num(input_material{2}); %[mm]
        G=str2num(input_material{3}); %[MPa]
        Ms_max_xmax=str2num(input_material{4}); %[N*mm]
        Ms_min_xmax=str2num(input_material{5}); %[N*mm]
        Kt_max_xmax=str2num(input_material{6}); %[N*mm^2]
        Kt_min_xmax=str2num(input_material{7}); %[N*mm^2]
        n_max=ceil(sqrt(str2num(input_material{8})));
        n_min=ceil(sqrt(str2num(input_material{9})));
        fprintf('n = [%f;%f] \n',n_min,n_max);
        nww_max=3*Ms_max_xmax/(t*mu*P); % [mm^2]
```

```
        nww_min=3*Ms_min_xmax/(t*mu*P); % [mm^2]
        fprintf('n*w^2 = [%f;%f] [mm^2] \n',nww_min,nww_max);
        nnnw_max=3*Kt_max_xmax/(t^3*G); % [mm]
        nnnw_min=3*Kt_min_xmax/(t^3*G); % [mm]
        fprintf('n^3*w = [%f;%f] [mm] \n',nnnw_min,nnnw_max);
        prompt_res = {'Enter selected w [mm]:','Enter selected
number of layers n[]:'};
        dlg_title='input';
        num_lines=1;
        def={'',''};

input_res=inputdlg(prompt_res,dlg_title,num_lines,def);
        w_res=str2num(input_res{1}); % [mm]
        n_res=str2num(input_res{2}); % adim
        w=w_res;
        n=n_res;
        h=n_res*t;
    end

    Kt_eff=G*w*h^3/3;
    r_eff=n^2;
    Kt_eff_0=Kt_eff/r_eff;
    fprintf('Kt = %f N x mm^2 \n',Kt_eff);
    fprintf('Kt_0 = %f N x mm^2 \n',Kt_eff_0);
    fprintf('r = %f \n',r_eff);
    Ms_eff=1/3*P*w^2*h*mu;
    fprintf('Ms = %f N \n',Ms_eff);
    sel3 = menu('Evaluate other load cases
performances?','Yes','No');
     if sel3==1
            prompt_l = {'Enter lenght l [mm]'};
            dlg_title='input';
            num_lines=1;
```

```
            def={''};

input_l=inputdlg(prompt_l,dlg_title,num_lines,def);
            l=str2num(input_l{1}); %[mm]
            %Axial forces
            F_l=n*mu*P*l*w; % [N]
            fprintf('F_l = %f N \n',F_l);
            %Bending moment
            prompt_ni = {'Enter poisson ratio []:'};
            dlg_title='input';
            num_lines=1;
            def={''};

input_ni=inputdlg(prompt_ni,dlg_title,num_lines,def);
            ni=str2num(input_ni{1}); %adim
            E=G*2*(1+ni);
            Kb=E*w*h^3/12; % [N*mm^2]
            fprintf('Kb = %f N x mm^2 \n',Kb);
            %Transversal forces
            K=E*w*h^3/(4*l^3);
            Ft=3*P*mu*w*h/2;
            fprintf('K = %f N/mm \n',K);
            fprintf('Ft = %f N \n',Ft);
        else
        end
end
```