```
function [ T , P , rho ] = stdatm_HOOD_LIAM( h )
%Outputs atmospheric conditions at a given altitude
%   Uses for loops to calculate the conditions in each layer of the
%   atmosphere
if h <= 11000 %First gradient

    T = 288.16 + ( -6.5*10^-3 ) .* h ; %Calculates Temperature at any
 altitude in the first gradient layer
    P = 101.325 .* ( T ./ 288.16 ).^( -9.8 ./ ( 287 .*
 -6.5*10^-3 ) ); %Calculates Pressure in the first gradient layer
    rho = 1.225 .* ( T ./ 288.16 ).^(( -9.8 ./ ( 287 .* -6.5*10^-3)) -
 1 ) ; %Calculates Density at any altitude in the first gradient layer

elseif h <= 25000 %First isothermal

    [ T , P , rho ] = stdatm_HOOD_LIAM( 11000 ); %Calculates values at
 the bottom of first isothermal layer
    P = P * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 11000)); %Calculates Pressure in the first isothermal layer
    rho = rho * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 11000)); %Calculates Density in the first isothermal layer

elseif h <= 47000 %Second gradient

    [ T , P , rho ] = stdatm_HOOD_LIAM( 25000 ); %Calculates values at
 the bottom of second gradient layer
    T = T + ( 3*10^-3 ) * ( h - 25000 ) ; %Calculates Temperature at
 any altitude in the second gradient layer
    P = P * ( T / 216.66 )^( -9.8 / ( 287 * 3*10^-3 ) ); %Calculates
 Pressure in the secon gradient layer
    rho = rho * ( T / 216.66 )^(( -9.8 / ( 287 * 3*10^-3)) -
 1 ) ; %Calculates Density at any altitude in the second gradient
 layer

elseif h <= 53000 %Second isothermal

    [ T , P , rho ] = stdatm_HOOD_LIAM( 47000 ); %Calculates values at
 the bottom of second isothermal layer
    P = P * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 47000)); %Calculates Pressure in the second isothermal layer
    rho = rho * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 47000)); %Calculates Density in the second isothermal layer

elseif h <= 79000 %Third gradient

    [ T , P , rho ] = stdatm_HOOD_LIAM( 53000 ); %Calculates values at
 the bottom of third gradient layer
    T = T + ( -4.5*10^-3 ) * ( h - 53000 ) ; %Calculates Temperature
 at any altitude in the third gradient layer
    P = P * ( T / 282.66 )^( -9.8 / ( 287 *
 -4.5*10^-3 ) ); %Calculates Pressure in the third gradient layer
```

```
    rho = rho * ( T / 282.66 )^(( -9.8 / ( 287 * -4.5*10^-3)) -
 1 ) ; %Calculates Density in the third gradient layer

elseif h <= 90000 %Third isothermal

    [ T , P , rho ] = stdatm_HOOD_LIAM( 79000 ); %Calculates values at
 the bottom of third isothermal layer
    P = P * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 79000)); %Calculates Pressure in the third isothermal layer
    rho = rho * (exp(1))^(( -9.8 / ( 287 * T )) * ( h -
 79000)); %Calculates Density in the third isothermal layer

elseif h <= 100000 %Fourth gradient

    [ T , P , rho ] = stdatm_HOOD_LIAM( 90000 ); %Calculates values at
 the bottom of third gradient layer
    T = T + ( 4*10^-3 ) * ( h - 90000 ) ; %Calculates Temperature at
 any altitude in the third gradient layer
    P = P * ( T / 165.66 )^( -9.8 / ( 287 * 4*10^-3 ) ); %Calculates
 Pressure in the third gradient layer
    rho = rho * ( T / 165.66 )^(( -9.8 / ( 287 * 4*10^-3)) -
 1 ) ; %Calculates Density in the third gradient layer

end


end
```

```
Not enough input arguments.

Error in stdatm_HOOD_LIAM (line 5)
if h <= 11000 %First gradient
```

Published with MATLAB® R2017a