```
%% Creates Object with static methods to define functions in a class

classdef set_1
        methods(Static)

    %% taylor function fpr hyperbolic cosine
     % @param take the input value and desired error
     % @return the vomputed value of x
      function[n] = taylor_cosh(x, err)
        res = 0;
        diff = 1;
        n = 0;
        expected = cosh(x);
        while (diff > err)
            res = res + ((x)^(2*n) ) / (factorial(2*n));
            n = n + 1;
           diff = abs(expected - res);

         end
      end
```

OUTPUT: 10

```
%% Fib Recurrsion Definition
    function[N] = fib_rec(N)
        if (N < 3)
                N = 1;
        else

         N = set_1.fib_rec(N-1) + set_1.fib_rec(N-2);
        end
    end
```

BEST TIME :1.682 seconds

```
function[j, fiblis] = fib_list(N)
        fiblis(1) = 1;
        fiblis(2) = 1;
        for n=1: (N - 2)
                fiblis(n + 2) = fiblis(n + 1) + fiblis(n);
        end

    j = fiblis(N);
end
```

BEST TIME: .0001032 seconds

```
        %% Calculates the ratio
         % returns 1, unless it is requested with two or more terms
         % to me the function is not zero, but one given the definition
         % provided
function[sig] = ratio(N)
             [~, seq] = (set_1.fib_list(N));
            sig = 1;
            for n=1:(N - 1)
                        sig = sig + ( (-1)^(n+1) ) / (seq(n) * seq(n+1));
            end

end
end
end
```