
ECES T580 In class Lecture 3 - Tyler Bradley

```
% 1. Create a function to compute fibonacci sequence
fib(13)

% 2. Calculate levensthein distance between two strings
lev_dist('kitten', 'sitting')

function output = fib(n)
    % create a vector of length n for the output
    out_vec = repelem(0, n);

    % define the first "next" value of the sequence
    next = 1;

    % add the first value of the sequence -> 0
    out_vec(1) = 0;
    % for loop that goes from 2 (as in second element of the output
    vector)
    % and first adds the "next" value to the output sequence and then
    % calculates the new "next" number by adding the current "next"
    number to
    % the last element in the out_vec
    for i = 2:n
        out_vec(i) = next;
        next = out_vec(i-1) + next;
    end
    %return the out_vec
    output = out_vec;

end

function output = lev_dist(a, b)
% define a and b as characters
a = char(a);
b = char(b);
% calculate the length of each character
len_a = length(a);
len_b = length(b);
%create a vector of zeros that is len_a+1 x len_b+1 dimensions
% we add the 1 to each length so that the first row and column will
remain
% zeroes in the calculation
my_mat = zeros(len_b+1, len_a+1);

% create a nested for loop that first loops over each row and then
each
% column of each row
for i = 2:len_b+1
    for j = 2:len_a+1
```

```

        % define the cost of a given diagonal move based on whether the
        % current cell of the matrix corresponds to a match in the two
words
        if a(j-1) == b(i-1)
            cost = 0;
        else
            cost = 1;
        end
        % calculate the potential values of the current cell by
calculating
        % what the value would be for a horizontal move (hor), a
vertical
        % move (ver), and a diagonal move (diag)
        ver = my_mat(i-1, j) + 1;
        hor = my_mat(i, j-1) + 1;
        diag = my_mat(i-1, j-1) + cost;

        % define the current cell as the minimum of the possible options
        my_mat(i, j) = min([hor, ver, diag]);

    end

end
% return the bottom right most cell
output = my_mat(len_b+1, len_a+1);
end

ans =

    0     1     1     2     3     5     8    13    21    34    55
89   144

ans =

    3

```

Published with MATLAB® R2018b