```
function [quo, rem] = PolyDivGF2(dvd, dvs, GF)
%POLYDIVGF2 does polynomial division in GF(2^m)
%   -dvd is a power form vector that represents the dividened
 polynomial
%   -dvs is a power form vector that represents the divisor polynomial
%   -GF is a list of cells containing the enumeration of the GF(2^m)
%   -quo is a return value that also is a power form vector that
 represents
%    the quotient polynomial
%   -rem is a return valuse that also is a power form vector that
%    represents the remainder polynomial

m = size(GF{1},2);
n = 2^m;
%degree of dividened and divisor
deg_dvd = size(dvd, 2) - 1;
deg_dvs = size(dvs, 2) - 1;
deg_quo = deg_dvd-deg_dvs;
%error handling
if(deg_dvd < 1)
    error("Degree of dividened too small, function is for polynomial
 division in GF(2^%d)\n\", m);
end

quo = zeros(1, deg_quo+1);
quo(:) = -1;

if(deg_dvd >= deg_dvs) %must have larger degree than divisor
    %loop through quotient array and set values
    for i = 1:deg_quo+1
        %loop through dividend and determine coef of quotient at the
        %current power
        quo(i) = DivGF2(dvd(i),dvs(1), GF);
        %determine the size of the power used to modify dividend and
 create
        %a polynomial containing the current quotient term (coeff and
 x
        %power)
        pow = (deg_quo+1) - i;
        temp = zeros(1, pow+1);
        temp(:) = -1;
        temp(1) = quo(i);
        %multiply that polynomial by the divisor and add padding in
 order
        %to modify dividend
        mult = PolyMultGF2(dvs, temp, GF);
        %resize mult/pad beginning with -1
        temp = zeros(1, deg_dvd+1);
        temp(:) = -1;
        temp(1,end-size(mult,2)+1:end) = mult;
        mult = temp;
        %subtract mult to dvd to modify dividend(adding is subtracting
 in GF(2^m)
        for j = 1:deg_dvd+1
            dvd(j) = AddGF2(dvd(j), mult(j), GF);
        end
        %loop until all quotient terms are filled
    end
    %setting remainder
    if(dvd(:) == -1) %if all coef of the dividend are now a^inf, there
 is no remainder
        rem = -1;
    else
        %otherwise the remainder is whatever is left in the dvd
        idx = find(dvd ~= -1);
        idx = idx(1);
        rem = dvd(1, idx:end);
    end
%set output when the divisor is bigger than the dividened
else
    rem = dvd;
    quo = -1;
end


end
```

Published with MATLAB® R2018b