```
function field = GenerateGF2(m, prim_poly, prnt_flag)
%{
GENERATEGF2: Used to generate enumeration of elements within GF(2^m)
 using an irriducible primitive polynomial of GF(2)[x].
INPUTS:
    m - the extension field parameter -> GF(2^m)
    prim_poly - The power form irriducible primitive polynomial of
 GF(2)[x]
    prnt_flag - optional parameter used for printing the enumerated
 field
OUTPUT:
    field - an array of cells with dimensions (2^m,1) whose elements
 are
          the rectangular form representation of polynomial elements
 in
          GF(2^m). The indices of the array correspond to the power of
          alpha (one of the 2^m symbols)that the polynomial element is
          equivalent to in GF(2^m)[x]. For an index (i) in the field
          array, i = e+2 where, e is some exponent of alpha (a^e) that
          exists in GF(2^m)[x]. a^inf symbol is represented as e=-1
 and
          is saved at field{1} (i=1) as [0 0 0 0].
%}

%handle optional parameter
if ~exist('prnt_flag','var')
    prnt_flag = false;
end

p = 2;
n = p^m - 1; %codeword length
syms a;
%create enumeration array of 2^m cells containing [0 0 0 0]
field = mat2cell(zeros(n+1, m), ones(1,n+1), m);
%do algebra, set highest degree term equal to the rest of the
 prim_poly
a_exp_m = prim_poly(2:end);
%create a polynomial GF(2)[x] polynomial that can increase the degree
 of the
%GF(2^m)[x] symbol by 1 when multiplied, ex: a^n * a^1 = a^(n+1)
a_1 = zeros(1, m);
a_1(end-1) = 1;

%generate equivalency matrix where cell index = power + 2 and -1 ==
 inf
for pow = -1:n-1
    if(pow == -1) %set a^inf mapping
        field{pow+2} = zeros(1, m);
    elseif(pow < m) %set mapping for symbols that are already
 represented in a_exp_m
        z = zeros(1, m);
        z(pow+1) = 1;
        field{pow+2} = flip(z);
    elseif(pow == m) %enumeration for a^m is the rest of prim_poly or
 a_exp_m
        field{pow+2} = a_exp_m;
    else %otherwise the symbol doesn't exist in GF(2)[x] and so must
 be put in terms of GF(2)[x] elements
        %multiply last element by a^1
        curr_cell = conv(field{pow+1},a_1);
        curr_cell = mod(curr_cell, 2); %be sure to mod
        if(curr_cell(end-m) == 1) %if the multiplication created an
 a^m symbol,
            %substitute the enumeration for a^m which should cancel
 some
            %terms through GF(2)[x] addition
            curr_cell = xor(curr_cell(1, end-(m-1):end), a_exp_m);
            field{pow+2} = double(curr_cell);
        else
            %otherwise, set the symbol = to the new polynomial of
 correct
            %size
            field{pow+2} = curr_cell(1, end-(m-1):end);
        end
    end

    %print out the equivalencies in power form
    if(prnt_flag)
        if(pow ~= -1)
            fprintf("a^%d = %s\n", pow, poly2sym(field{pow+2},a));
        else
            fprintf("a^inf = %s\n", poly2sym(field{pow+2},a));
        end
    end
end

end
```

Published with MATLAB® R2018b