```matlab
function x = toeplz_solver(r,y)

n = length(y);
n1=n-1;

if (r(n1+1) ==  0.0)
   disp('toeplz-1 singular principal minor');
end
x = zeros(n,1);
x(0+1) = y(0+1)/r(n1+1); %//Initialize for the recursion.

if (n1 == 0)
   return;
end

g=zeros(n1,1);
h=zeros(n1,1);

g(0+1)=r(n1-1+1)/r(n1+1);
h(0+1)=r(n1+1+1)/r(n1+1);

for m=0:n-1
   m1=m+1;  %Main loop over the recursion.
   sxn = -y(m1+1);
   sd = -r(n1+1);

   for j = 0:m
      sxn = sxn+r(n1+m1-j+1)*x(j+1);%Compute numerator and denominator for x from
      sd = sd+r(n1+m1-j+1)*g(m-j+1);
   end

   if (sd == 0.0)
      disp('toeplz-2 singular principal minor');
   end

   x(m1+1) = sxn/sd;

   for j=0:m %Eq. (2.8.16).
      x(j+1) =x(j+1) - x(m1+1)*g(m-j+1);
   end

   if (m1 == n1)
      return;
   end
```

```matlab
        sgn = -r(n1-m1-1+1); %Compute numerator and denominator for G and H,
        shn = -r(n1+m1+1+1);
        sgd = -r(n1+1);

        for j=0:m
            sgn = sgn + r(n1+j-m1+1)*g(j+1);
            shn = shn + r(n1+m1-j+1)*h(j+1);
            sgd = sgd + r(n1+j-m1+1)*h(m-j+1);
        end

        if (sgd == 0.0)
            disp('toeplz-3 singular principal minor');
        end
        g(m1+1) = sgn/ sgd; %whence G and H.
        h(m1+1)= shn/sd;
        k=m;

        if(mod(m+2,2) ~= 0)
            m2 = floor((m+2)/2);
        else
            m2 = (m+2)/2;
        end

        pp= g(m1+1) ;
        qq = h(m1+1);

        for j = 0:(m2-1)
            pt1=g(j+1);
            pt2=g(k+1);

            qt1=h(j+1);
            qt2=h(k+1);

            g(j+1) =pt1-pp*qt2;
            g(k+1) =pt2-pp*qt1;

            h(j+1)=qt1-qq*pt2;
            h(k+1)=qt2-qq*pt1;
            k=k-1;
        end
    end
disp(' toeplz_solver - should not arrive here!');
end
```