

```

fprintf('No.5(c)\n\n');
close all;
clear all;
% Write a code that solves the linear system
%function f(x)
f = @(x) -2*x.^0;

beta = 0; gama = 2/3;
n = 100;

h = 1/(n+1);

j = [1:n]';
xj = j*h;

%RHS b
b1 = (h^2)*f(xj(1)) - ((2*gama)/h);
jj = [2:n-1]'; bj = (h^2)*f(xj(jj));
bn = (h^2)*f(xj(n)) - beta;
b = [b1;bj;bn];

%tridiagonal matrix A
aa = -2; bb = 1; cc = 1;
A = diag(aa*ones(1,n)) + diag(bb*ones(1,n-1),1) + diag(cc*ones(1,n-1),-1);
A = sparse(A);

%vectors u and v
u = zeros(n,1); u(1) = 1;
v = zeros(n,1);
for i = 1:n
    v(i) = 2;
end

p = algorithm(A,u,v,b);
%x1 = (A - u*v')\b
px = @(x) 1 - x.^2;

%plotting
plot(px(xj),'--*')
hold on
plot(p,'o')
legend('p(x) = 1-x^2','soln p');
xlabel('n');ylabel('x');
title('Agraph of x against n')

fprintf('According to the plot above the solution converges to p(x) = 1 - x^2 as n tends to infinity\n\n');

function p = algorithm(A,u,v,b)
    %solve Az = b
    z = A\b;
    %solve Ay = b
    y = A\u;
    %compute alpha
    alpha = v'*y;
    %compute beta
    beta = v'*z;
    %compute x
    if alpha == 0
        exit
    else
        p = z + (beta/(1-alpha)).*y;
    end
end

```

No.5(c)

According to the plot above the solution converges to $p(x) = 1 - x^2$ as n tends to infinity

