

4.

```

1 function x = gauss_pp(A, b)
2
3 %no. of rows of A
4 n = size(A, 1);
5
6 for k=1:n-1
7     %partial Pivoting
8     [~, p] = max(abs(A(k:n, k)));
9     p = p + k - 1;
10    if p ~= k
11        %row swapping
12        A([k p], :) = A([p k], :);
13        b([k p]) = b([p k]);
14    end
15    %forward elimination
16    for i = k+1:n
17        m = A(i,k) / A(k,k);
18        A(i, k:n) = A(i, k:n) - m * A(k, k:n);
19        b(i) = b(i) - m * b(k);
20    end
21 end
22 x = zeros(n, 1);
23 %backward substitution
24 for i = n:-1:1
25     x(i) = (b(i) - A(i,i+1:n)*x(i+1:n)) / A(i,i);
26 end
27 return
28 end

```

Listing 1: Function for Gaussian elimination with partial pivoting (*gauss_pp.m*).

```

1 clear all; clc;
2
3 %inputs
4 n = 100;
5 A = diag(ones(n,1)) + diag(ones(n-1,1),1) + diag(ones(n-1,1),-1);
6 b = zeros(n, 1);
7 b(50) = 1;
8 %Gauss elimination with partial pivoting
9 x = gauss_pp(A, b);
10 %built in
11 x_ = A \ b;
12 %plot
13 f = figure(1);
14 plot(1:n, x, "bo-", LineWidth=1.5, MarkerSize=8, DisplayName="Gauss elimination w/
    partial pivoting"); hold on;
15 plot(1:n, x_, "rx", LineWidth=1.5, MarkerSize=9, DisplayName="Matlab built-in"); hold
    off;
16 set(gca,"TicklabelInterpreter","latex","FontSize", 15);
17 legend("boxoff");
18 legend("Location", "northwest", "interpreter", "latex");
19 xlabel("$j$", "interpreter", "latex");
20 ylabel("$x_j$", "interpreter", "latex");
21 pbaspect([3 1 1]);
22 exportgraphics(f,"lin_solv.pdf");

```

Listing 2: Input-output script for Gaussian elimination with partial pivoting (*input_output.m*).

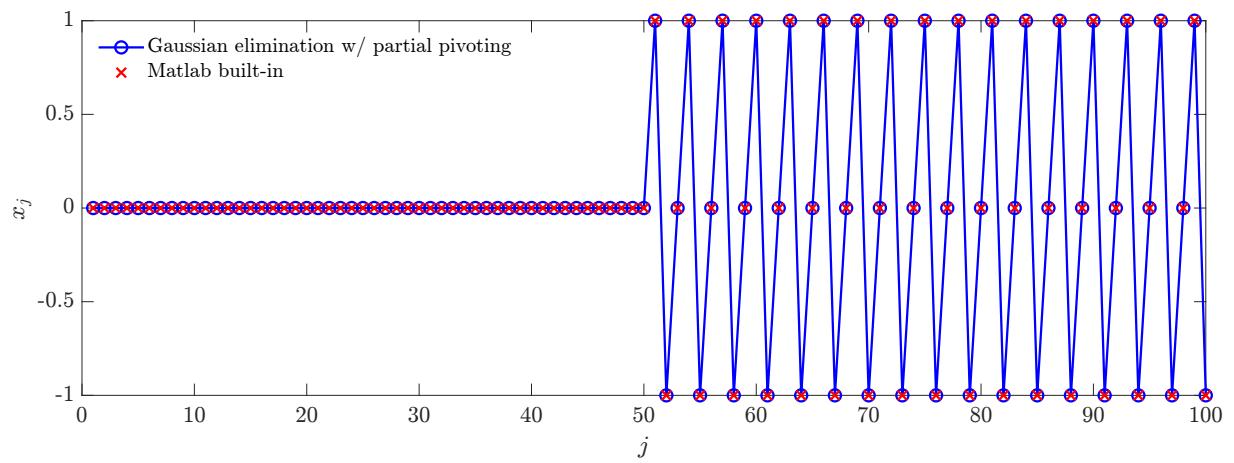


Figure 1: Plot of \vec{x} .