

```
N_blocks = 1000;

N = 100;
p = [0.001 0.999]; % probability of transmission success transmission error
P = [0 cumsum(p)] ;
Transmission = zeros(N,1); % vector to hold results of each Transmission
% (the memory is "pre-allocated" in this
% way to speed up Matlab processing

Transmission_succes = zeros(1,2);

for k = 1:N_blocks
    for i=1:N % loop over number of Transmission
        x=rand(1,1);
        for j=2:length(P), % determine result of each Transmission
            if( (P(j-1)<x) & (x<P(j)) )
                Transmission(i) = j-1;
                break
            end
        end
    end
end
%4.B
N = 10000;

% Probability of transmission success and transmission error
p = [0.999 0.001];
P = [0 cumsum(p)];

% Vector to hold results of each transmission
Transmission = zeros(N, 1);

% Loop over number of transmissions
for i = 1:N
    x = rand(1, 1);
    % Determine result of each transmission
    for j = 2:length(P)
        if P(j-1) < x && x < P(j)
            Transmission(i) = j - 1;
            break;
        end
    end
end

Prob_signal = zeros(1,length(p));

for l=1:length(p)
    Prob_signal(1,l) = length(find(Transmission==l));
end

if Prob_signal(1,1) >= 2
    Transmission_succes(1) = Transmission_succes(1) + 1;
```

```
else
    Transmission_succes(2) = Transmission_succes(2) + 1;
end

end

Transmission_succes/N_blocks
Prob_signal = zeros(1, length(p));
for i = 1:length(p)
    Prob_signal(1, i) = sum(Transmission == (i - 1)) / N;
end

disp(Prob_signal);

%4.D
N = 100000;
p = 0.001;

n_less_2_error = 0;
n_given_less_2_error = 0;

for i = 1:N
    bits = rand(1, 100) < (1 - p);
    num_errors = sum(~bits);

    if num_errors < 2
        n_less_2_error = n_less_2_error + 1;

        if num_errors == 0
            n_given_less_2_error = n_given_less_2_error + 1;
        end
    end
end

result = n_given_less_2_error / n_less_2_error;
disp([1-result, result])
```