```
% Computer Lab 5
% Simulating Brownian motion via a random walk

T = 1;
N = 100;% number of grid points
M = 1000; % number of paths
h = T/N; % stepsize
B = zeros(N,M);
for j = 1:M
    for i = 1:N-1
        B(i+1,j) = B(i,j) + sqrt(h)*randn();
    end
end
% In the matrix B each column is a Brownian path on [0,T] so B(1,1) =
% B_0(omega_1), B(1,2) = B_0(omega_2), etc.
% Likewise B(nh,1) = B_nh(omega_1) for each n in {0, 1, ..., N}

% Part (a.)
% Take t = 1
answer = mean(B(N,:)); % approximating E[B_1] by averaging of B_1 over
 M samples
disp(['E[B_1] should equal zero; E[B_1] is approximately
 ',num2str(answer)])
fprintf('\n');
% B(N,:) selects all columns in the N-th row of B, i.e the N-th row of
 B
answer = var(B(N,:)); % approximating Var(B_1)
disp(['Var[B_1] should equal 1; Var[B_1] is approximately
 ',num2str(answer)])
fprintf('\n');

% If I wanted to compute E[B_t] for any t I could take mean(B(round(t/
h),:))
% because T = Nh so n = t/h, i.e. B_t = B(t/h,:)

% Part (b.)
% Take t = 1, s = 1/2
covariance_matrix = cov(B(N,:),B(N/2,:));
answer = covariance_matrix(1,2); % approximates Cov(B_1,B_1/2)
% The answer here should be min(t,s) = min(1,1/2) = 1/2
disp(['Cov[B_1,B_0.5] should equal 1/2; Cov[B_1,B_0.5] is
 approximately ',num2str(answer)])
fprintf('\n');
% The answer here should be min(t,s) = min(1,1/2) = 1/2

% Part (c.)
% Take t = 1
answer = mean(max(B));
true = sqrt(2/pi);
% The true value here is given by sqrt(2/pi)
% The answer here should be min(t,s) = min(1,1/2) = 1/2
```

```
disp(['E[M_1] should equal ',num2str(true),'; E[M_1] is approximately
 ',num2str(answer)])
```

```
E[B_1] should equal zero; E[B_1] is approximately -0.026843

Var[B_1] should equal 1; Var[B_1] is approximately 1.0394

Cov[B_1,B_0.5] should equal 1/2; Cov[B_1,B_0.5] is approximately
 0.50437

E[M_1] should equal 0.79788; E[M_1] is approximately 0.73009
```

Published with MATLAB® R2015a