The stock price model is the classical geometric Brownian motion model:

$$S_T = S_0 \exp\left\{\left(r - \frac{1}{2}\sigma^2\right)T + \sigma W_T\right\}$$

In this code, we can simulate the price of the butterfly option by

(a) control variate method

(b) stratification with k = 100.

```
r = 0.05;
sigma = 0.2;
S0 = 50;
K1 = 45;
K2 = 50;
K3 = 55;
T = 1;
n = 10000;

[v1, se1] = control_variate(r, sigma, S0, K1, K2, K3, T, n)
```

```
v1 = 0.9159
se1 = 0.0142
```

```

k = 100;
N = 10000;
[v2, se2] = stratification(r, sigma, S0, K1, K2, K3, T, k, N)
```

```
v2 = 0.9190
se2 = 4.5138e-04
```

Functions

(a) Control Variate.

```
function [v, se] = control_variate(r, sigma, S0, K1, K2, K3, T, n)
    X = zeros(1, n);
    Y = zeros(1, n);
    S = zeros(1, n);
    for i = 1:n
        Z = normrnd(0,1);
        S(i) = S0*exp((r-1/2*sigma^2)*T + sigma*sqrt(T)*Z); % S_T
        X(i) = exp(-r*T)*(max(S(i)-K1, 0) + ...
            max(S(i)-K3, 0) - 2*max(S(i)-K2, 0));
        Y(i) = exp(-r*T)*S(i); % Control Variable Y = e^(-rT)*S_T
    end
    X_bar = 1/n * sum(X);
    Y_bar = 1/n * sum(Y);
    b_hat = sum((X - X_bar).*(Y - Y_bar)) / ...
            sum((Y - Y_bar).^2); % Estimate Optimal Coefficient
```

```
    H = X - b_hat*(Y - S0); % We know E[Y]=S0
    v = 1/n * sum(H);
    se = sqrt(1/(n*(n-1)) * (sum(H.^2) - n*v^2));
end
```

(b) Stratification with k = 100.

```
function [v, se] = stratification(r, sigma, S0, K1, K2, K3, T, k, N)
    n = zeros(1, k);
    for i = 1:k
        n(i) = 1/k * N; % use proportional allocation, p_i=1/k
    end
    mu = zeros(1, k);
    s = zeros(1, k);
    for i = 1:k
        h_of_U = zeros(1, n(i));
        for j = 1:n(i)
            V = rand;
            U = (i-1)/k + V/k;
            S = S0*exp((r-1/2*sigma^2)*T + sigma*sqrt(T)*norminv(U));
            h_of_U(j) = exp(-r*T)*(max(S-K1, 0) + ...
                        max(S-K3, 0) - 2*max(S-K2, 0));
        end
       mu(i) = 1/n(i) * sum(h_of_U);
       s(i) = sqrt(1/(n(i)-1) * sum((h_of_U - mu(i)).^2));
    end
    v = 1/k * sum(mu);
    se = 1/k * sqrt(sum((1./n).*(s.^2)));
end
```