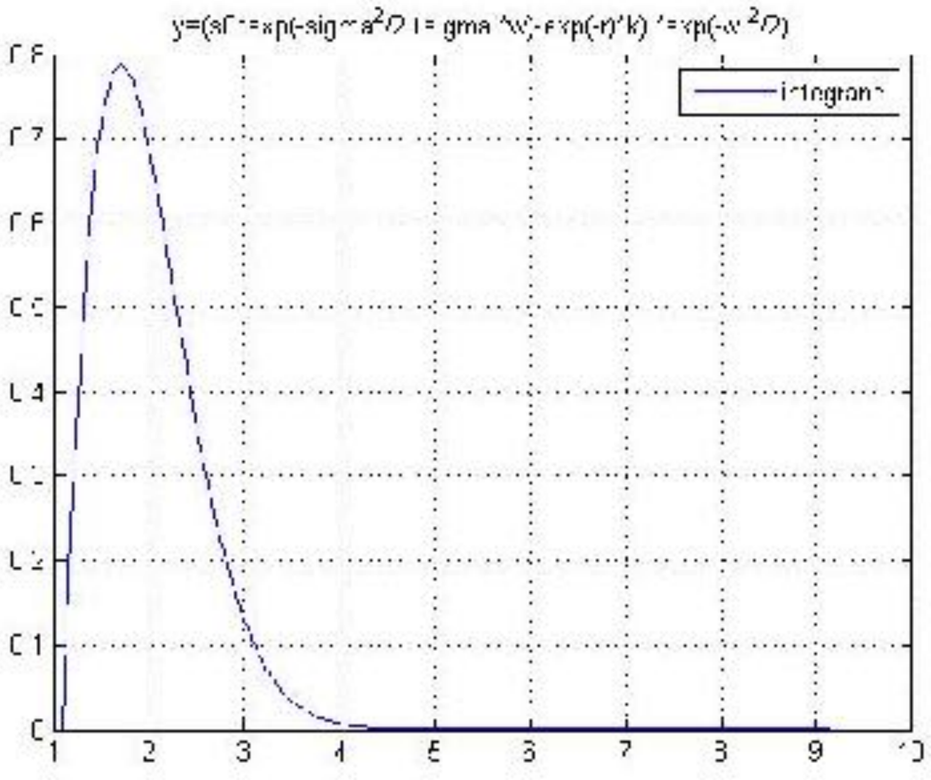
Homework2
Group2
Member: Qian Lou, Kai Yan

Assignment 1

(a) Using "assignment1" below

```
function [] = assignment1a
s0 = 30;
r = 0.03;
sigma = 0.15;
k = 36;
d1=(log(s0/k)+r-sigma^2/2)/sigma;
w=linspace(-d1,10,100);
y=(s0*exp(-sigma^2/2+sigma.*w)-exp(-r)*k).*exp(-w.^2/2);
plot(w,y)
title('y=(s0*exp(-sigma^2/2+sigma.*w)-exp(-r)*k).*exp(-w.^2/2)');
legend('integrand')
grid on
```



From the graph above, we can easily tell that the integrand increseses from -d_=1.0905 to 1.7043 and decreases from 1.7043 to infinity. The peak of this graph occurs at w=1.7043.

For part b) and part c) Using "Assignment1c"

```
function [] = assignment1c
s0 = 30;
mu = 0.03;
sigma = 0.15;
```

```
K = 36;
n = 1000;
m = 500;
d1=(log(s0/K)+mu-sigma^2/2)/sigma;
upper=exp(d1);
bs = s0*normcdf((log(s0/K)+mu+sigma^2/2)/sigma)-exp(-mu)*K*normcdf((log(s0/
K)+mu-sigma^2/2)/sigma);

% assignment 1b
intv=(upper-0)/n;
y=0:intv:exp(d1)-intv; %using 1000 values;
y=y+intv/2;
intg=1./(sqrt(2*pi)*y).*(s0*exp(-sigma^2/2-sigma*log(y))-exp(-mu)*K).*exp(-
(log(y).^2)/2);
numerical=mean(intg)*exp(d1)

%Crude Monte Carlo
x=rand(n,m)*exp(d1);
g=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
estimate=mean(g)*(exp(d1)-0);
Crude_bias= mean(estimate)-bs;
Crude_std = std(estimate);
Crude_MSE = (mean(estimate)-bs)^2+var(estimate);

%Important Sampling
alpha = 2;
beta = 2.6992;
x = betarnd(alpha,beta,n,m)*upper;
g1=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
g2= betapdf(x/upper,alpha,beta)./upper;
estimate = mean(g1./g2); % Importance sampling estimate
Importance_bias = mean(estimate)-bs;
Importance_std = std(estimate);
Importance_MSE = (mean(estimate)-bs)^2+var(estimate);

% Stratified Sampling
x(1:300,:) = rand(300,m)*0.1; % draw 300 values from U[0,0.1]
x(301:800,:) = rand(500,m)*0.1+0.1; % draw 500 values from U[0.1,0.2]
x(801:1000,:) = rand(200,m)*(upper-0.2)+0.2; % draw 200 values from
U[0.2,upper]
g=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
estimate = 0.1*mean(g(1:300,:))+0.1*mean(g(301:800,:))+
(upper-0.2)*mean(g(801:1000,:));
Stratified_bias = mean(estimate)-bs;
Stratified_std = std(estimate);
Stratified_MSE = (mean(estimate)-bs)^2+var(estimate);

% Control Variate
x = rand(n,m)*upper;
ratio1=3334/1245;
ratio2=2.0521/1.7778;
h=ratio2*betapdf(x.*ratio1,2,3);
hvalue=(6*upper^2-8*ratio1*upper^3+3*ratio1^2*upper^4)*ratio1*ratio2;
g=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
diff =g-h; % calculate g(x)-h(x)
estimate = upper*mean(diff)+hvalue;
Control_bias = mean(estimate)-bs;
Control_std = std(estimate);
Control_MSE = (mean(estimate)-bs)^2+var(estimate);
```

```
% Antithetic Variate
x(1:371,:) =  rand(371,m)*0.1245; % draw 371 values from U[0,0.1245]
x(372:1000,:) =  rand(629,m)*(upper-0.1245)+0.1245; % draw 629 values from
U[0.1235,upper]
g1=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
x(1:371,:) = 0.1245-x(1:371,:); % mirror image of x over the range
[0,0.1245]
x(372:1000,:) = upper+0.1245-x(372:1000,:); % mirror image of x over the
range [0.1245,upper]
g2=1./(sqrt(2*pi)*x).*(s0*exp(-sigma^2/2-sigma*log(x))-exp(-mu)*K).*exp(-
(log(x).^2)/2);
estimate = 0.1245*mean(g1(1:371,:)+g2(1:371,:))/2+...
    (upper-0.1245)*mean(g1(372:1000,:)+g2(372:1000,:))/2;
% Std1 = sqrt(var((g1(1:300,:)+g2(1:300,:))/2)/300+...
    %9^2*var((g1(301:1000,:)+g2(301:1000,:))/2)/700)
    % calculate standard deviation when m = 1, block the 2 lines for m > 1
Antithetic_bias = mean(estimate)-bs;
Antithetic_std = std(estimate);
Antithetic_MSE = (mean(estimate)-bs)^2+var(estimate);

[Crude_bias, Crude_std ,Crude_MSE;
Importance_bias,Importance_std,Importance_MSE;
Stratified_bias,Stratified_std,Stratified_MSE ;
Control_bias,Control_std,Control_MSE ;
Antithetic_bias,Antithetic_std,Antithetic_MSE]
```

(b) We will be able to reach the result, which gives us the numerical integration equals 0.3921. This perfectly matches the price calculated from the B-S option pricing model.

(c) First, we did the a variable transformation by letting $y = e^{-w}$, and obtain a finite range integral.
(1) For crude Monte Carlo, 1000 values are sampled from the range *U*(0,0.3361).
(2) For importance sampling, 1000 values are sampled from *Beta*(2, 2.6992) to match the peak when x=0 .1245, since the peak for *Beta*(α, β) occurs at $(\alpha - 1)/(\alpha - 1)(\beta - 1)$.
(3) For stratified sampling, after looking at the original integrand graph, we decided to smaple 300 values from *U*[0, 0.1], 500 values from *U*[0.1, 0.2], and 200 values from *U*[0.2, 0.3361].
(4) For control variate, choose $h(w)$ = *Beta*(2, 3) ×0.5606 and $\int h(w)dw$ = 0.3146/1.0950 = 0.5606.
(5) For antithetic variate, 371 values are sampled from *U*[0,0.1245] and 629 values from *U*[0.1245, 0.3361], since the numerical integrand reaches the peak when x is the 371st value with the range [0.0,3361]

(d) After running the code above, we will get the result listed in the spreadsheet below. We can see all variance reduction methods are unbiased and reduced variance significantly. To be specific, the control variate and the stratified sampling give us the best result.

Method	**Bias**	***Std***	***MSE***
Crude Monte Carlo	0.0004	0.0074	0.0001
Importance	0.0001	0.0026	0.0000

Stratified	-0.0001	0.0063	0.0000
Control	-0.0000	0.0015	0.0000
Stratified	-0.0000	0.0009	0.0000

Assignment 2

Using the routine "Assignment2" below

```
% Price European call option with strike K via variance reduction methods

function [] = assignment2new
s0 = 30;
mu = 0.03;
sigma = 0.15;
K = 36;
bp = [40 45 50 55 60];
numb = 5;
bs = s0*normcdf((log(s0./K)+mu+sigma^2/2)/sigma)-exp(-
mu)*K.*normcdf((log(s0./K)+mu-sigma^2/2)/sigma);
    % black-scholes price
n = 64; % number of time intervals
m = 1000; % number of sample paths
rep = 500; % number of estimates for each variance reduction method
Estimate = zeros(6,numb); % rows are variance reduction methods; columns
are estimated prices for different strikes
MSE_Ratio = zeros(6,numb); % rows are variance reduction methods; columns
are MSE ratios for different strikes


for bb = 1:numb

crude = zeros(1,rep);
import = zeros(1,rep);
control_s = zeros(1,rep);
control= zeros(1,rep);
antithetic = zeros(1,rep);
strat = zeros(1,rep);

B = bp(bb); % current strike
d1=(log(K/s0)-mu+sigma^2/2)/sigma;
d = ones(n+1,m)/n; % d is the increment of time t
d(1,1:m) = zeros(1,m);
tt = cumsum(d); % tt is the sample path of time t

for ii = 1:rep
    w=bbridge(m,n);
    s = s0*exp((mu-sigma^2/2)*tt+sigma*w); % stock prices along each sample
path

    % Crude Monte Carlo
    pay = max(s(n+1,1:m)-K,0).*(max(s)<B);
    crude(ii) = mean(pay)/exp(mu); %discount to current value
    paycall=max(s(n+1,1:m)-K,0); %plain vanilla
```

```
    % Control Variate
    beta = regress(pay',s(n+1,1:m)'); % use stock price as control variate
    control_s(ii) = mean(pay/exp(mu)-beta*(s(n+1,1:m)/exp(mu)-s0));
    beta1 = regress(pay',paycall'); % use stock price as control variate
    control_c(ii) = mean(pay/exp(mu)-beta1*(paycall/exp(mu)-bs));

    % Antithetic Variate
    w2 = -w;
    s = s0*exp((mu-sigma^2/2)*tt+sigma*w2);
    pay2 = max(s(n+1,1:m)-K,0).*(max(s)<B);
    antithetic(ii) = (mean(pay2)/exp(mu)+crude(ii))/2;

     %importance sampling
    theta=log(B/K);
    s = s0*exp((mu+theta-sigma^2/2)*tt+sigma*w);
    pay = max(s(n+1,1:m)-K,0).*exp(-(theta/sigma)*w(n+1,1:m)-(theta/
sigma)^2/2);
    pay=pay.*(max(s)<B);
    import(ii) = mean(pay)/exp(mu);

    % Stratified Sampling
    w = zeros(n+1,m);
    subsample = m/10;   % 10 strata for balanced w(n+1,1:m) = randn(1,m)
    %lower, start
    start=normcdf(d1);
    j = 0;
    for i = 1:10
        u = start+rand(1,subsample)*(1-start)/10+(i-1)*(1-start)/10;   %
draw subsample from Uniform
        w(n+1,j+1:j+subsample) = norminv(u);
        j = j+subsample;
    end
    h = log(n)/log(2); % generate Brownian bridge
    k = 1; % number of cells to fill at each layer
    for v = 1:h
        ip = power(2,h-v); % gap between cell to be filled and nearest
cells already filled
        for j = 1:k
            s = ip*(2*j-1)+1; % position of the cell to be filled
            u = s-ip; % lower position
            t = s+ip; % upper position
            w(s,1:m) = randn(1,m)*sqrt((t-u)/4/n)+(w(u,1:m)+w(t,1:m))/2; %
interpolation
        end
        k = k*2;
    end
    s = s0*exp((mu-sigma^2/2)*tt+sigma*w);
    pay = max(s(n+1,1:m)-K,0).*(max(s)<B);
    strat(ii) = mean(pay)/exp(mu)*(1-start);

end

Estimate(1,bb) = mean(crude);
Estimate(2,bb) = mean(control_s);
Estimate(3,bb) = mean(control_c);
Estimate(4,bb) = mean(antithetic);
Estimate(5,bb) = mean(import);
Estimate(6,bb) = mean(strat);

MSE_Ratio(1,bb) = var(crude)/var(crude);
MSE_Ratio(2,bb) = var(crude)/var(control_s);
MSE_Ratio(3,bb) = var(crude)/var(control_c);
MSE_Ratio(4,bb) = var(crude)/var(antithetic);
```

```
MSE_Ratio(5,bb) = var(crude)/var(import);
MSE_Ratio(6,bb) = var(crude)/var(strat);

end

Strike = K
BS_Price = bs
Estimate
MSE_Ratio
```

We will be able to get the following two charts below.

The first chart gives us the means of estimate value for each method with varies barriers

Method	***B* = 40**	***B* = 45**	***B* = 50**	***B* = 55**	***B* = 60**
Crude Monte Carlo	0.1111	0.3203	0.3803	0.3889	0.3913
Control_stock	0.1110	0.3203	0.3802	0.3890	0.3912
Control_call	0.1114	0.3204	0.3820	0.3908	0.3919
Antithetic	0.1120	0.3202	0.3809	0.3902	0.3937
Important	0.1118	0.3164	0.3807	0.3922	0.3917
Stratified	0.1117	0.3179	0.3805	0.3907	0.3922

The second chart is the variance ratios of crude Monte Carlo against other methods.

Method	***B* = 40**	***B* = 45**	***B* = 50**	***B* = 55**	***B* = 60**
Crude Monte Carlo	1	1	1	1	1
Control_stock	1.0317	1.0663	1.0745	1.0753	1.0809
Control_call	1.1018	2.0860	11.4639	72.4023	359.7478
Antithetic	2.1972	2.0088	2.4489	2.0662	1.8956
Important	2.4522	8.1796	15.1260	7.0871	2.5171
Stratified	19.1441	34.8009	115.4754	192.7645	181.0095

Conclusion:

1). For the most cases, importance sampling works the best except when the barrier is too high to be hit, the control variance method using European call as the control variate works better.

2). When the barrier is too high to be hit, the European call option could be approximately equal to the barrier option. Therefore, using European call option as control variate reduced a lot of variance and turn out to be the best methods.