```
# lab4.R                        script file for lab4 calculations
#
# author: Eric Zivot
# created: September 17, 2008
# revised: July 6, 2012
#


#
# Matrix algebra
#

# part a)
matA = matrix(c(1,4,7,2,4,8,6,1,3), 3, 3, byrow=T)
matA
matB = matrix(c(4,4,0,5,9,1,2,2,5), 3, 3, byrow=T)
matB
vecx = matrix(c(1,2,3), 3, 1)
vecx
vecy = matrix(c(5,2,7), 3, 1)
vecy

# part b)
t(matA)
t(matB)
t(vecx)
t(vecy)

# part c)

matA + matB
matA - matB
2*matA
matA%*%vecx
t(vecy)%*%matA%*%vecx

# d) x + y = 1, 2x + 4y = 2
# 1st line: y = 1 - x; 2nd line: y = 0.5 - 0.5 x
x.vals = seq(-1, 2, length = 20)
y.vals = 1 - x.vals
plot(x.vals, y.vals, type="l", col="blue", lwd=2)
abline(a=0.5,b=-0.5, lwd=2)
abline(v=1)
abline(h=0)

matA = matrix(c(1,1,2,4), 2, 2, byrow=T)
vecb = matrix(c(1,2), 2, 1)
matA
vecb
matA.inv = solve(matA)
matA.inv
z = matA.inv%*%vecb
z

# e) portfolio problem

vecmu = matrix(c(0.01,0.04,0.02), 3, 1)
matSigma = matrix(c(0.1,0.3,0.1,0.3,0.15,-0.2,0.10,-0.20, 0.08), 3, 3, byrow=T)
vecx = matrix(c(1/3,1/3,1/3), 3, 1)
vecmu
```

```
matSigma
vecx

crossprod(vecmu, vecx)
t(vecmu)%*%vecx
crossprod(vecx, matSigma%*%vecx)
t(vecx)%*%matSigma%*%vecx


#
# VI simulate time series data
#

# simulate MA(1) process with theta > 0
ma1.model.5 = list(ma=0.5)
mu = 0.05
set.seed(123)
ma1.sim.5 = mu + arima.sim(model=ma1.model.5, n=250,
                           innov=rnorm(n=250, mean=0, sd=0.1))
acf.ma1.model.5 = ARMAacf(ma=0.5, lag.max=10)

par(mfrow=c(3,1))
        ts.plot(ma1.sim.5, main="MA(1) Process: mu=0.05, theta=0.5",
                xlab="time",ylab="y(t)")
        abline(h=0)
        plot(1:10, acf.ma1.model.5[2:11], type="h", col="blue", main="theoretical ACF")
        tmp=acf(ma1.sim.5, lag.max=10, main="Sample ACF")
par(mfrow=c(1,1))

# do the same for the the MA model with theta = 0.9

# simulate AR(1) process with phi > 0
ar1.model.5 = list(ar=0.5)
mu = 0.05
set.seed(123)
ar1.sim.5 = mu + arima.sim(model=ar1.model.5, n = 250,
                           innov=rnorm(n=250, mean=0, sd=0.1))
acf.ar1.model.5 = ARMAacf(ar=0.5, lag.max=10)

par(mfrow=c(3,1))
        ts.plot(ar1.sim.5,main="AR(1) Process: mu=0.05, phi=0.5",
                xlab="time",ylab="y(t)")
        abline(h=0)
        plot(1:10, acf.ar1.model.5[2:11], type="h", col="blue", main="Theoretical ACF")
        tmp=acf(ar1.sim.5, lag.max=10, main="Sample ACF")
par(mfrow=c(1,1))

# do the same for the model with phi = 0.9
```