Homework 6 Solution
PHZ 5156, Computational Physics
September 27, 2005

PROBLEM 1(a)

Code

```
from scipy import *
from LinearAlgebra import *

a = array( (
    (2.5000000e+00,   2.8867513e-01,  -2.0412415e-01,   3.5355339e-01),
    (2.8867513e-01,   3.5000000e+00,   3.5355339e-01,  -6.1237244e-01),
    (-2.0412415e-01,   3.5355339e-01,   1.5000000e+00,  -8.6602540e-01),
    ( 3.5355339e-01,  -6.1237244e-01,  -8.6602540e-01,   2.5000000e+00)))

b = array( (
    (7.5000000e+00,   2.8867513e-01,  -2.0412415e-01,   3.5355339e-01),
    (2.8867513e-01,   7.8333333e+00,   1.1785113e-01,  -2.0412415e-01),
    (-2.0412415e-01,   1.1785113e-01,   3.4166667e+00,  -2.4537386e+00),
    (3.5355339e-01,  -2.0412415e-01,  -2.4537386e+00,   6.2500000e+00)))

c = array( (
    (2.5000000e+00,   2.8867513e-01,   4.0824829e-01,  -7.0710678e-01),
    (2.8867513e-01,   2.1666667e+00,  -4.7140452e-01,   8.1649658e-01),
    (4.0824829e-01,  -4.7140452e-01,   2.5833333e+00,   7.2168784e-01),
    (-7.0710678e-01,   8.1649658e-01,   7.2168784e-01,   1.7500000e+00)))

print "a=\n",a
print "b=\n",b
print "c=\n",c
```

Output

```
a=
[[ 2.5         0.28867513 -0.20412415  0.35355339]
 [ 0.28867513  3.5         0.35355339 -0.61237244]
 [-0.20412415  0.35355339  1.5        -0.8660254 ]
 [ 0.35355339 -0.61237244 -0.8660254   2.5       ]]
b=
[[ 7.5         0.28867513 -0.20412415  0.35355339]
 [ 0.28867513  7.8333333   0.11785113 -0.20412415]
 [-0.20412415  0.11785113  3.4166667  -2.4537386 ]
 [ 0.35355339 -0.20412415 -2.4537386   6.25      ]]
c=
[[ 2.5         0.28867513  0.40824829 -0.70710678]
 [ 0.28867513  2.1666667  -0.47140452  0.81649658]
 [ 0.40824829 -0.47140452  2.5833333   0.72168784]
 [-0.70710678  0.81649658  0.72168784  1.75      ]]
```

Agrees with assignment.

PROBLEM 1(b)

Code fragment

```
#Problem 1(b)
evalsa, evecsa = eigenvectors(a)
for n in arange(4):
        print "eigenvalue",n,,":",round(evalsa[n],5)
        print "eigenvector",n,":",round(evecsa[n],5),"\n"
```

Output

```
eigenvalue 0 : 3.0
eigenvector 0 : [ 0.70711  0.40825 -0.28868  0.5    ]

eigenvalue 1 : 2.0
eigenvector 1 : [ 0.70711 -0.40825  0.28868 -0.5    ]

eigenvalue 2 : 4.0
eigenvector 2 : [ 0.      -0.8165  -0.28868  0.5    ]

eigenvalue 3 : 1.0
eigenvector 3 : [-0.      -0.      -0.86603 -0.5    ]
```

PROBLEM 1(c)

Code fragment

```
#Problem 1(c)
S = transpose(evecsa)
print "S=\n",round(S,5),"\n"

Sdagger = conjugate(transpose(S))
aprime = dot(dot(Sdagger,a),S)
bprime = dot(dot(Sdagger,b),S)
cprime = dot(dot(Sdagger,c),S)

print "A'=\n",round(aprime)
print "B'=\n",round(bprime)
print "C'=\n",round(cprime),"\n"

print "AB-BA=\n",round(dot(a,b)-dot(b,a),5)
print "AC-CA=\n",round(dot(a,c)-dot(c,a),5)
```

Output (i)

```
S=
[[ 0.70711  0.70711  0.      -0.     ]
 [ 0.40825 -0.40825 -0.8165  -0.     ]
 [-0.28868  0.28868 -0.28868 -0.86603]
 [ 0.5     -0.5      0.5     -0.5    ]]
```

Output (ii)

```
A'=
[[ 3.  0. -0.  0.]
 [ 0.  2.  0. -0.]
 [-0.  0.  4. -0.]
 [ 0. -0. -0.  1.]]
B'=
[[ 8.  0.  0. -0.]
 [ 0.  7. -0. -0.]
 [ 0. -0.  8.  0.]
 [-0. -0.  0.  2.]]
C'=
[[ 2.  0. -1. -0.]
 [ 0.  3.  0.  0.]
 [-1.  0.  1. -0.]
 [-0.  0. -0.  3.]]
```

Diagonal (A', B')

Not Diagonal (C')

Output (iii)
AB-BA=
```
[[ 0. -0.  0. -0.]
 [ 0.  0.  0. -0.]
 [-0. -0.  0. -0.]
 [ 0.  0.  0.  0.]]
```

AC-CA=
```
[[ 0.      -0.57735 -0.20412  0.35355]
 [ 0.57735  0.      -0.35355  0.61237]
 [ 0.20412  0.35355  0.      -0.     ]
 [-0.35355 -0.61237  0.       0.     ]]
```

Matrices A and B commute, so eigenvectors of one are also eigenvectors of the other. That is, they can be "simultaneously diagonalized."

Matrix C does not commute with matrix A, and therefore does not share the same eigenvectors.

PROBLEM 1(d)

Matrix B' is already diagonalized. Its eigenvalues can be read off the diagonal: 8, 7, 8, 2. Matrix C' can be written in block diagonal form by interchanging states 2 and 3 (i.e., swapping column 2 with column 3 and row 2 with row 3). This gives

C''=
```
[[  2.  -1. 0.  0.]
 [ -1.  1.  0.  0.]
 [  0.  0.  3.  0.]
 [  0.  0. 0.  3.]]
```

The lower right entries give two eigenvalues, 3 and 3 again. The other two eigenvalues come from diagonalizing the upper-left-hand block of C'':

$$\det\begin{pmatrix} 2-\lambda & -1 \\ -1 & 1-\lambda \end{pmatrix} = 0 = (2-\lambda)(1-\lambda) - 1 = \lambda^2 - 3\lambda + 1$$

$$\Rightarrow \lambda = \frac{1}{2}\left(3 \pm \sqrt{5}\right) = 2.6180, 0.3820$$

PROBLEM 1(e)

Of course I started with the diagonal matrices *A'*,*B'* and the block diagonal *C'*, set up a unitary matrix *S*, and then applied similarity transformations: $A=SA'S^\dagger$, $B=SB'S^\dagger$, and $C=SC'S^\dagger$.

PROBLEM 2(a)

Matrix T is nmax by nmax, with entries

$$T_{nm} = \left\langle u_n \middle| \hat{T} u_m \right\rangle = 2\int_0^L dx \sin\left(\frac{(n+1)\pi x}{L}\right)\left(-\frac{d^2}{dx^2}\right)\sin\left(\frac{(m+1)\pi x}{L}\right)$$

$$= \left[\frac{(m+1)\pi}{L}\right]^2 2\int_0^L dx \sin\left(\frac{(n+1)\pi x}{L}\right)\sin\left(\frac{(m+1)\pi x}{L}\right) = \left[\frac{(m+1)\pi}{L}\right]^2 \left\langle u_n \middle| u_m \right\rangle$$

$$= \left[\frac{(m+1)\pi}{L}\right]^2 \delta_{nm}$$

Thus with n,m=0,1,2,3,4, matrix T is

$$T = \left(\frac{\pi}{11}\right)^2 \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 9 & 0 & 0 \\ 0 & 0 & 0 & 16 & 0 \\ 0 & 0 & 0 & 0 & 25 \end{pmatrix}$$

PROBLEM 2(b)

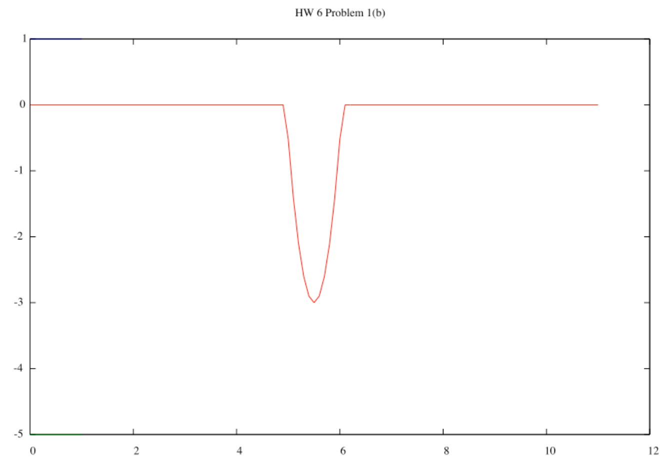
Problem 2(b) Code

```
from scipy import *
import Gnuplot,Gnuplot.funcutils

#Set up the constants
L = 11.
w = L/10.
V0 = 3.
nmax = 5

#Calculate and plot V(x)
h = 0.1
x = arange(0.,L+h,h)
Vhat = 0*x
for j in arange(len(x)):
	xj = x[j]
	if abs(xj-L/2)<w/2:
		Vhat[j] = V0 * ( (4./(w**2))*(xj-L/2)**2 - 1)


g = Gnuplot.Gnuplot(debug=0)
g.title('HW 6 Problem 1(b)')
g('set data style lines')
g1 = Gnuplot.Data(x,Vhat)
g2 = Gnuplot.Data((-5.,-5.))
g3 = Gnuplot.Data((1.,1.))
g.plot(g1,g2,g3)
```



PROBLEM 2(c)

To be a linear operator requires

- Closure: Is Vf in the Hilbert space if f(x) is in the Hilbert space? Obviously yes; the result is a function defined on [0,L] and satisfies the boundary conditions.
- Linearity: V(αf+βg)=αf(x)+βg(x) is obviously true.

PROBLEM 2(d)

In this basis V becomes a matrix with elements V_{nm} where

$$V_{nm} = \left\langle u_n \middle| \hat{V} u_m \right\rangle = 2\int_0^L dx \sin\left(\frac{(n+1)\pi x}{L}\right) V(x) \sin\left(\frac{(m+1)\pi x}{L}\right)$$

The integrals could be done analytically, but I asked you to do them numerically for practice.

Problem 2(d) code fragment

```
#Calculate the matrix elements of V by quadrature using the
#trapezoidal rule
V = zeros((nmax,nmax),Float)
for  n in arange(nmax):
        for m in arange(n+1):
                un = sqrt(2./L)*sin((n+1)*pi*x/L)
                um = sqrt(2./L)*sin((m+1)*pi*x/L)
                Vnm = 0.5*un[0]*Vhat[0]*um[0]
                Vnm = Vnm + 0.5*un[-1]*Vhat[-1]*um[-1]
                for j in arange(1,len(x)-1):
                        Vnm = Vnm + un[j]*Vhat[j]*um[j]
                Vnm = h*Vnm
                V[n,m] = Vnm
                if (m<>n): V[m,n] = conjugate(Vnm)
print "V=\n",round(V,5)
```

I reduced the step size to h=0.0001 for greater accuracy.

```
V=
[[-0.39965 -0.       0.3917   0.      -0.37616]
 [-0.      -0.00794  0.       0.01554 -0.     ]
 [ 0.3917   0.      -0.3841  -0.       0.36923]
 [ 0.       0.01554 -0.      -0.03042  0.     ]
 [-0.37616 -0.       0.36923  0.      -0.35564]]
```

PROBLEM 2(e)

The operators do not commute, and therefore the matrices do not either:

$$\left(\hat{V}\hat{T}-\hat{T}\hat{V}\right)f(x)=V(x)\left(-\frac{d^2}{dx^2}\right)f(x)-\left(-\frac{d^2}{dx^2}\right)\left[V(x)f(x)\right]$$
$$=-Vf''+\frac{d}{dx}\left(V'f+Vf'\right)=-Vf''+V''f+2V'f'+Vf''$$
$$=V''+2V'f'$$

```
VT-TV=
[[ 0.       0.      -0.2556  -0.       0.73638]
 [-0.       0.      -0.      -0.01521  0.     ]
 [ 0.2556   0.       0.       0.      -0.48187]
 [ 0.       0.01521 -0.       0.      -0.     ]
 [-0.73638 -0.       0.48187  0.       0.     ]]
```