Homework 2: QR Factorization
Due: February 7, 2014

1. A **projector** is a square matrix P that satisfies
$$P^2 = P$$
and is an **orthogonal projector** if additionally $P = P^T$.

   (a) Prove $I - P$ is a projector if P is a projector. Additionally, $I - P$ is an orthogonal projector if P is an orthogonal projector.

   (b) Prove
   - Null(P)=Range($I - P$)
   - Range(P)=Null($I - P$)
   - Null(P)$\cap$Null($I - P$) $= \{0\}$

   Thus you have just proven that a projector P separates R^n into two subspaces, i.e. given $v \in \mathsf{R}^n$ there exists x such that
   $$v = Px + (I - P)x.$$

   (c) The QR decomposition, creates a sequence of orthogonal vectors $\{q_1, q_2, \ldots, q_k\}$ at the k-th step that span the first k columns of A. Prove that
   $$P = Q_k Q_k^T$$
   is an orthogonal projector that projects a vector orthogonally onto the space spanned by the first k-columns of A where $Q_k = [q_1, \ldots, q_k]$. What does this imply about where $I - Q_k Q_k^T$ projects?

   (d) Prove
   $$\arg\min_{x \neq 0} \|Ax - b\|_2^2 = \arg\min_{x \neq 0} \|Rx - Q^T b\|_2^2$$
   where $A = QR$ is the reduced QR Factorization on an $m \times n$ matrix A. Thus solving the least squares problem is equivalent to solving
   $$Ax = QQ^T b \Leftrightarrow Rx = Q^T b$$
   It may be helpful to remember that $QQ^T \neq I$, but $[Q\hat{Q}][Q\hat{Q}]^T = I$ where $A = [Q\hat{Q}]\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right]$ is the full QR Factorization of A. Also $\left\| \left[\begin{smallmatrix} Q^T \\ \hat{Q}^T \end{smallmatrix}\right] y \right\|_2 = \|y\|_2$

   Note: here we assume that the columns of A are linearly independent, so R in $A = QR$ is invertible.

2. G&C: 7.16

3. MATLAB: Create three programs to compute the QR factorization using

   - classical Gram-Schmidt,
   - modified Gram-Schmidt,
   - Householder transformations

   Test your three methods on the Vandermonde matrix that can be created in Matlab using the code:

```
m=100;n=15;
t=linspace(0,1,m)';
% Create Vandermonde
A = [];
for i = 1:n
    A = [A t.^(i-1)];
end
```

   Comment on the orthogonality of your 3 corresponding Q matrices. Then test the method with the right-hand side being created by the following code:

```
x = ones(n,1);
b = A*x;
```

   Which method would you use in practice and why?