CS 421: Fall 1997

Solutions to Prelim

1. (a) True. C is clearly symmetric. If $x \neq 0$, then $x^TCx = y^Ty > 0$, where $y = Ax \neq 0$ since A has full rank. (b) False. The smallest positive number in a base-β floating point system $f(\beta, t, L, U)$ depends on the lower bound L for the exponent. Assuming the IEEE standard is met, the relative error of a floating point arithmetic operation is bounded by the unit rounding error $\frac{1}{2}\beta^{-(t-1)}$ where t is the precision. (c) True. $x = A^Tb$ which can be computed in $O(n^2)$ flops. (d) False. The accuracy of the computed solution depends on both the stability of the algorithm and the condition of the problem. If the solution of $Ax = b$ is computed stably, the relative residual will be small. However, if the condition number of the matrix A is large, the computed solution may only have a few (or even zero) correct digits. (e) True. If A is singular then at some stage all elements on and below the pivot position will be zero (in exact arithmetic). Thus this column is already in triangular form and computation can go straight on to the next stage.

2. The Cholesky factor L is zero everywhere except on the diagonal and the last row:

$$A = \begin{bmatrix} d_1 & & & & e_1 \\ & d_2 & & & e_2 \\ & & \ddots & & \vdots \\ & & & d_{n-1} & e_{n-1} \\ e_1 & e_2 & \cdots & e_{n-1} & d_n \end{bmatrix}, \qquad L = \begin{bmatrix} f_1 & & & & \\ & f_2 & & & \\ & & \ddots & & \\ & & & f_{n-1} & \\ g_1 & g_2 & \cdots & g_{n-1} & f_n \end{bmatrix},$$

where $f_i = \sqrt{d_i}$, $g_i = e_i/\sqrt{d_i}$, for $i = 1, \ldots, n-1$, and

$$f_n = \sqrt{d_n - e_1^2/d_1 - \ldots - e_{n-1}^2/d_{n-1}}.$$

The following Matlab function overwrites d and e with f and g:

```
function [d,e]=arrowchol(d,e)
n=length(d);
for i=1:n-1
    d(i)=sqrt(d(i));
    e(i)=e(i)/d(i);
    d(n)=d(n)-e(i)^2;
end
d(n)=sqrt(d(n));
```

Or you can avoid the loop by vectorizing the code (this version assumes both d and e are column vectors):

```
function [d,e]=arrowchol(d,e)
```

```
n=length(d);
d(1:n-1)=sqrt(d(1:n-1));
e=e./d(1:n-1);
d(n)=sqrt(d(n)-e'*e);
```

There are 4 floating point operations per loop, so the leading term in the flop count is $4n$.

3. To solve

$$\min_{x_1\in\Re^n,\ldots,x_p\in\Re^n} ||Ax_1 - b_1||_2 + \cdots + ||Ax_p - b_p||_2$$

we minimize each term independently, i.e., for $j = 1, \ldots, n$ we solve

$$\min_{x_j\in\Re^n} ||Ax_j - b_j||_2 = \min_{x_j\in\Re^n} ||Rx_j - Q^T b_j||_2.$$

The solution can be computed by solving $\tilde{R}x_j = v_j$, where $\tilde{R}$ is $n \times n$ upper triangular and v_j consists of the n first entries of $Q^T b_j$

$$R = \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix}, \qquad Q^T b_j = \begin{bmatrix} v_j \\ w_j \end{bmatrix}.$$

The minimum value is $||w_j||_2$.

Given Q, R, and assuming the vectors $b_1, \ldots, b_p$ are stored as the columns of the $m\times p$ matrix B, the following Matlab segment solves the problem (and computes the minimum value).

```
X=zeros(n,p);
minval=0;
for j=1:p
    y=Q'*B(:,j);
    X(:,j)=R(1:n,:)\y(1:n);      % backsubstitution
    minval=minval+norm(y(n+1:m));
end
```

Now the solution is given by the columns of X. In Matlab it can also be computed simply as

```
V=Q(:,1:n)'*B;
X=R(1:n,:)\V;   % multiple backsubstitutions
```