CS 4220: Prelim 2 Solutions

Problem 1	20 points	8.7
Problem 2	20 points	15.2
Problem 3	20 points	11.5
Problem 4	20 points	9.8
Problem 5	10 points	7.8
Problem 6	10 points	4.7

57.7

90–100	xx
80-89	xx
70-79	xxxxx
65-69	xxxxxxxx
60-64	xxxxxx
55-59	xxxxxxxx
50-54	xxxxx
45-49	xxxx
40-44	xxxxxxxx
35-39	xx
30-34	xxxx

A (65-100), B (50-60), C (35-40), median = 58.

1. Assume that A has the following form:

$$A = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & 0 & \times & \times \\ \times & 0 & 0 & \times \\ \times & 0 & 0 & 0 \\ \times & 0 & 0 & 0 \\ \times & 0 & 0 & 0 \end{bmatrix}.$$

Using the "$\times - 0$" notation, show how this matrix can be reduced to upper triangular form by premultiplying it with a sequence of Givens rotations. The first step is illustrated. (You may or may not have to use all the grids to display the reduction.)

$$\begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & 0 & \times & \times \\ \hline \times & 0 & 0 & \times \\ \hline \times & 0 & 0 & 0 \\ \hline \times & 0 & 0 & 0 \\ \hline \times & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & 0 & \times & \times \\ \hline \times & 0 & 0 & \times \\ \hline \times & 0 & 0 & 0 \\ \hline \times & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & 0 & \times & \times \\ \hline \times & 0 & 0 & \times \\ \hline \times & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & 0 & \times & \times \\ \hline \times & 0 & 0 & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow$$

$$\begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow$$

$$\rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \times & \times & \times & \times \\ \hline 0 & \times & \times & \times \\ \hline 0 & 0 & \times & \times \\ \hline 0 & 0 & 0 & \times \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \; & \; & \; & \; \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|}\hline \; & \; & \; & \; \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline \end{array} \rightarrow$$

6 points if you just reduce th efirst column and leave everything else alone

Roughly 13 points if you show that you know how Givens rotations work and how to apply them but don't reach the upper triangular form

2. We wish to solve the blocked $Ax = b$ system

$$\begin{bmatrix} A_1 & C & F \\ C^T & A_2 & H \\ F^T & H^T & A_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

where it is assumed that the matrix of coefficients is symmetric and positive definite and that all the blocks are large, square, and sparse. **(a)** Given an approximate solution

$$x^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{bmatrix},$$

specify the three linear systems that must be solved if $x^{(k+1)}$ is determined using a block Gauss-Seidel update.

15 points:

$$\begin{aligned} A_1 x_1^{(k+1)} &= b_1 - C x_2^{(k)} - F x_3^{(k)} \\ A_2 x_2^{(k+1)} &= b_2 - C^T x_1^{(k+1)} - H x_3^{(k)} \\ A_3 x_3^{(k+1)} &= b_3 - F^T x_1^{(k+1)} - H^T x_2^{(k+1)} \end{aligned}$$

-2 for Jacobi instead of Gauss-Seidel

(b) Describe the factorizations that should be used to solve these systems and the conditions that they should be satisfy if the overall block Gauss-Seidel procedure is to be effective.

5 points:

Sparse Cholesky: $P_i A_i P_i^T = G_i G_i^T$ and the G_i should be sufficiently sparse.

Intelligent talk about spectrtal radius OK too

3. Assume the availability of the following function that can be used to compute an approximate Schur decomposition:

```
  function [V,D] = JacobiCyclic(A,tol)
% Cyclic Jacobi algorithm for the symmetric Schur decomposition.
% A is nxn and symmetric
% tol is a positive real number
% V is nxn and orthogonal
% D is nxn and diagonal
% V'*A*V = D + E where norm(E,'fro') <= tol*norm(A,'fro')
```

(a) Given a value for `tol`, how would you compute approximate Schur decompositions for both $A \in \mathbb{R}^{n\times n}$ and $A + zz^T$ where $z \in \mathbb{R}^n$ and $\| zz^T \|_1 << \| A \|_F$? Make effective use of `JacobiCyclic`.

10 Points:

```
[VA,DA] = JacobiCyclic(A,tol);
w = VA'*z;
[U,DB] = JacobiCyclic(DA + w*w',tol)
% The off(input matrix) is small since z (and w)  is small in norm
VB = VA*U;
%  VA'*A*VA = DA   is an approximate Schur decomposition for A
%  VB'*(A+zz')*VB = DB   is an approximate Schur decomposition for A+zz'
```

5 points for Schur of A

2 points if youplay with z in an intelligent way

(b) Suppose $A \in \mathbb{R}^{n\times n}$ and $1 \leq p < q \leq n$ and that $J \in \mathbb{R}^{n\times n}$ is a Jacobi rotation with

$$J([\,p\ q\,],[\,p\ q\,]) \;=\; \begin{bmatrix} c & s \\ -s & c \end{bmatrix}$$

How many flops are required for the update $A \leftarrow AJ$? Show work. Exploiting symmetry, how would you then compute the update $A \leftarrow J^T A$?

10 points

```
B = A([p q],[p q]);

A(:,[p q]) = A(:,[p q])*[c s ; -s c]   % 6n

% Basically no extra flop work to get complete update...
A([p q],:) = A(:,[p q])';
A([p q],[p q]) = [c s;-s c]'*B*[c s ; -s c]     % O(1)
```

O(n) OK instead of 6n

-2 or 3 points if update OK but do not exploit symmetry

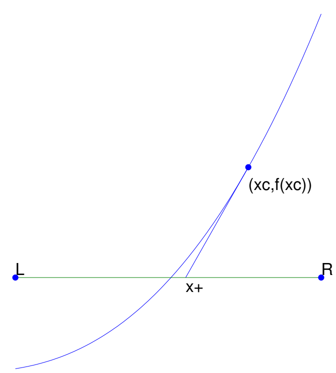
4. Assume that $f(x)$, $f'(x)$, and $f''(x)$ are defined everywhere. It is known that $[L, R]$ is a bracketing interval and

- $f'(x) \geq \alpha > 0$ for all $x \in [L, R]$.
- $f''(x) > 0$ for all $x \in [L, R]$.

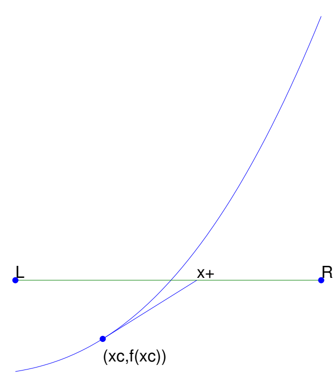
Prove by picture and simple algebra that Newton's method will converge if the starting value is selected from $[L, R]$ and

$$|f(L)| \leq \alpha \cdot (R - L).$$

If the starting value is to the right of the root, then the Newton iterates are monotone decreasing and always to the right of the root:



If the starting value is to the left of the root, then the next iterate is to the right of the root:



We must guarantee that $x_+ = x_c - f(x_c)/f'(x_c) \leq R$. The picture tells us that x_+ gets bigger as and x_c gets smaller. Thus we must have

$$L - f(L)/f'(L) \leq R$$

i.e., $|f(L)| = -f(L) \leq (R - L)f'(L)$ But this follows from the assumption that $|f(L)| \leq \alpha(R - L)$.

7 points for correct picture of f

7 points for situation when x_c is to right of root (5 for tangent line, 2 for saying the iterates are monotone decreasing and are never to the left of th eroot

6 points for the situation when the current iterate is to the left. (4 for tangent line that shows next iterate to the right, 2 points for discussion of when the next iterate is in [L,R].

5. The call `[Q,R,P] = qr(A,0)` returns a matrix $Q \in \mathbb{R}^{10\times 4}$ with orthonormal columns, a permutation matrix

$$P = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

and an upper triangular matrix $R \in \mathbb{R}^{4\times 4}$ with the property that $AP = QR$. If A is very close to a rank-3 matrix, what can you say about R? What column of A is almost in the span of the other columns? Why?

r_{44} will be very small so

$$\begin{aligned} A(:,1) = AP(:,4) &= r_{14}Q(:,1) + r_{24}Q(:,2) + r_{34}Q(:,3) + r_{44}Q(:,4) \\ &\approx r_{14}Q(:,1) + r_{24}Q(:,2) + r_{34}Q(:,3) \end{aligned}$$

Since the first three columns of Q span the same subspace as the first three columns of AP, it follows that $A(,:)$ is almost in the span of $A(:,3)$, $A(:,4)$, and $A(:,2)$.

6. Assume that function `F(t,alpha)` has been implemented that returns the value of

$$F(t,\alpha) \;=\; \int_\alpha^t g(z)dz$$

where $g(z)$ is defined everywhere and is always positive. The following script can be used to compute t_* so that $F(t_*,0) = F(1,0)/2$:

```
ave = F(1,0)/2;
L = 0; fL = 0;
R = 1; fR = F(1,0);
ave = fR/2;
while R-L> 2^(-30)
   Mid = (L+R)/2;
   fMid = F(Mid,0);
   if fMid<ave
      L = Mid; fL = fMid;
   else
      R = Mid; fR = fMid;
   end
end
tStar = (L+R)/2;
```

Assume that the cost of an $F(t,\alpha)$-evaluation in dollars is $|t-\alpha|$. (a) Give an upper bound for the cost of this method. You can leave your answer in summation form, e.g.

$$\text{upper bound} \;=\; \sum_k (\text{some fraction that depends on } k)$$

7 points:

Things get more and more expensive if Mid (and L) keep moving to the right. If the root is very close to one, then L is monotone increasing, i.e, $L = 0, 1/2, 3/4, ...$ Thus, the F evaluation cost would be

$$1 \,+\, \sum_{k=1}^{30}(1-1/2^k)$$

(b) Modify the line `fMid = F(mid,0)` to reduce the worst case cost upper bound to about two dollars. Do not modify any other line. Explain your answer.

3 points:

```
fMid = fL + F(Mid,L);
```

Now the cost summation is $1 + 1/2 + 1/4 + \cdots + 1/2^{30} \approx 2$.