# Assignment 4 Solutions

Aaron Cahn
University of Wisconsin-Madison
cahn@cs.wisc.edu

March 25, 2015

# 1 Solutions

## 1.1 Question 1

### 1.1.1 Part A

### 1.1.2 Part B

Listing 1: Matlab Commands

```
function [L,U] = lu_sym(U)
    tic
    % get the dimension of the input matrix
    m = size(U);
    % initialize L
    L = eye(m);

    % each row of U
    for k=1:m-1
        % each row below the current row (rows to modify)
        for i=k+1:m
            % create the elimination factor
            % grab the elimination factor from above the diagonal
            % U(k,i) instead of U(i,k)
            L(i,k)=U(k,i)/U(k,k);
            % update the upper triangular part of row i of U
            % start j at i (diagonal element)
            % process each j (from diagonal to end of row)
            for j=i:m
                U(i,j)=U(i,j)-L(i,k)*U(k,j);
            end
        end
    end
    % grab the upper triangular portion of U
    U = triu(U);
    toc
end
```

### 1.1.3 Part C

The complexity of Gauss elimination is is $O(\frac{m^3}{3})$. Regular gauss elimination works on an entire matrix (*i.e.,* touching all elements of the matrix) during *LU* factorization. However, for a symmetric matrix we have proven in part A that we only need to touch the upper triangular elements of a matrix. This constitutes half of the matrix. Therefore, the complexity of the symmetric *LU* factorization algorithm will be $\frac{1}{2}\frac{m^3}{3}$ or simply $O(\frac{m^3}{6})$.

### 1.1.4 Part D

## 1.2 Question 2

### 1.2.1 Part A

### 1.2.2 Part B

### 1.2.3 Part C

## 1.3 Question 3

### 1.3.1 Part A

### 1.3.2 Part B

### 1.3.3 Part C

### 1.3.4 Part D