# CSE636 Data Integration - Fall 2008

## Assignment #1 - Due Wednesday, September 17<sup>th</sup> – 12:00pm

Please email the instructor a soft copy of your solutions. Your files should be named as indicated in the problem statements.

## Problem 1 (20 points)
Consider the following DTD for XML documents:
```
<!DOCTYPE Univ [
        <!ELEMENT    University (Dept+)>
        <!ATTLIST    Dept Name ID #REQUIRED>
        <!ELEMENT    Dept (Faculty | Staff | Student)+>
        <!ATTLIST    Faculty Name ID #REQUIRED>
        <!ELEMENT    Faculty (Office, Salary)>
        <!ATTLIST    Staff Name ID #REQUIRED>
        <!ELEMENT    Staff (Office, Salary)>
        <!ATTLIST    Student Name ID #REQUIRED
                        Advisor IDREF #REQUIRED>
        <!ELEMENT    Student (Dorm?)>
        <!ELEMENT    Office (#PCDATA)>
        <!ELEMENT    Salary (#PCDATA)>
        <!ELEMENT    Dorm (#PCDATA)> ]
```
Specify an example XML document (**1a.xml**) that is valid with respect to the University DTD. Your example should include exactly one Department, one Faculty member, and two Student elements. Also, specify the smallest, in terms of element, attribute and text nodes, XML document (**1b.xml**) you can think of that is valid with respect to the University DTD.

## Problem 2 (30 points)
Write a DTD (**2.dtd**) and an XML Schema (**2.xsd**) which will define the following structure for documents of type **book**. A **book** has a **title**, a **date**, which consists of a **month** and a **year**, and one or more **chapter** elements. Each **chapter** has a list of either 4 or 5 **section** elements. Each **section** has an **introduction**, a list of **section** elements and a **conclusions** element, in that order. For each **section**, either the element **introduction** is present, or the **conclusions** element, or both.

## Problem 3 (30 points)
Write a DTD (**3.dtd**) and an XML Schema (**3.xsd**) which will define binary trees consisting of **red** and **black** nodes. The root of the tree is a **black** node. All leaves are **black** nodes. Every **red** node has two **black** child nodes. Every **black** node has two **red** child nodes, unless one of them or both are leaves, in which case they are **black**.

## Problem 4 (20 points)
We call a DTD satisfiable if there is a document that satisfies it. Show an example DTD (**4.dtd**) which is not satisfiable.