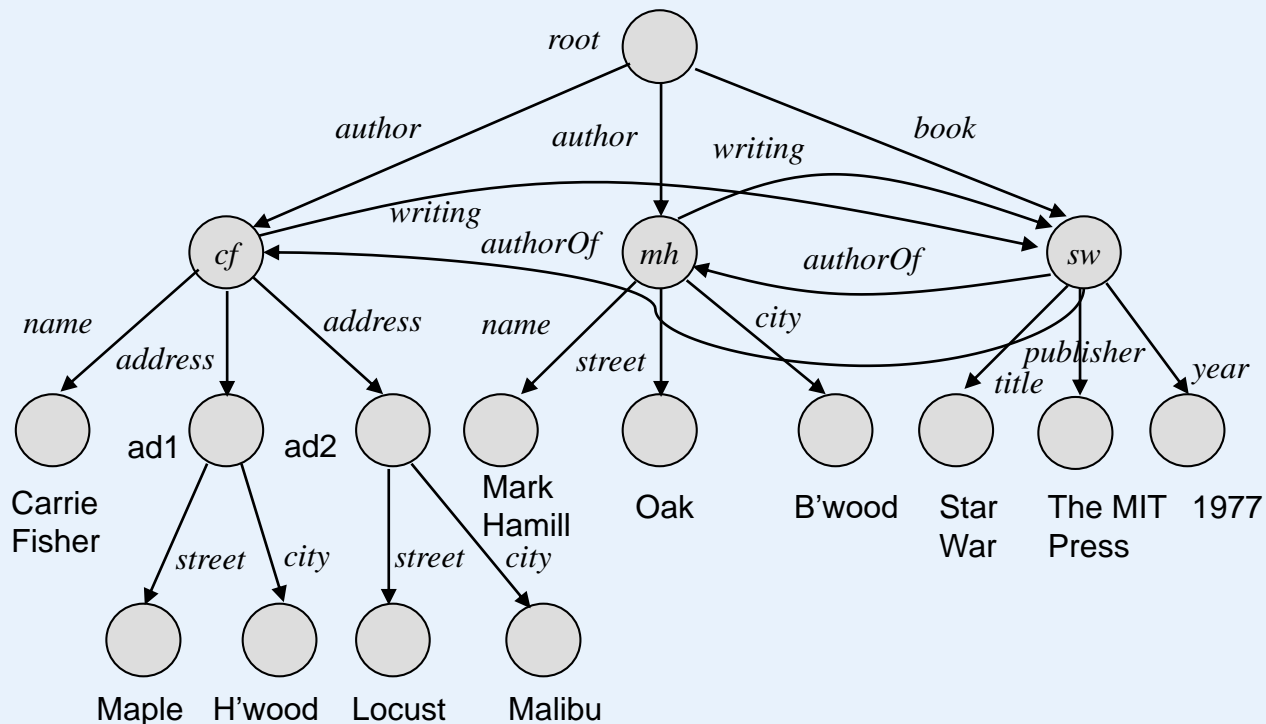


Assignment #2

Due: March 08, 2024

(sent to teaching assistant: Ms. R. Kondam, rasagnya53@gmail.com)

1. (15) Represent the following graph as an XML document.



Assignment #2

2. (15) The following is a DTD for books. Please produce an XML document conforming to the DTD.

```
<!DOCTYPE AuthorBook [  
  <!ELEMENT AuthorBook      (Author*, Book*)>  
  <!ELEMENT Author (Name, Address+)>  
    <!ATTLIST Author  
      AuthorId      ID          #REQUIRED  
      writing        INREFS     #IMPLIED  
    >  
  <!ELEMENT Name  (#PCDATA)>  
  <!ELEMENT Address (Street, City)>  
  <!ELEMENT Street (#PCDATA)>  
  <!ELEMENT City  (#PCDATA)>  
  <!ELEMENT Book  (Title, Publisher, Year)>  
    <!ATTLIST      Book  
      BookIn      ID          #REQUIRED  
      authorOf    IDREFS     #REQUIRED  
    >  
  <!ELEMENT Title  (#PCDATA)>  
  <!ELEMENT Publisher (#PCDATA)>  
  <!ELEMENT Year   (#PCDATA)>  
>
```

Assignment #2

3. (15) Define an XML-schema which is equivalent to the DTD shown in Question 2.
4. (25) Write an algorithm to transform a simplified XPath expression (in which the subexpressions in any condition can be connected only with \wedge , e.g., `/StarMovieData/Star[//City = "Malibu" and //Street = "123 Maple St."]/Name`) to a tree structure.
(Hint:
 1. First, generate a path P covering all the element names on the main path in XPath expression X (not including the element names in predicates.)
 2. Check each element e on P . If e is associated with a predicate: $x_1 = a_1 \wedge x_2 = a_2 \wedge \dots \wedge x_m = a_m$, make a series call of the algorithm recursively to generate a subtree for each x_i ($i = 1, \dots, m$).
 3. Make it clear how the subtrees generated for x_i 's are connected to P .)

Assignment #2

5. (20) The following is a DTD for a set of documents on books.
- (a) Write an FLWR expression to find all the books authored by D. Knuth.
 - (b) Write an FLWR expression to find all books published by Addison Wesley Longman in 2007.

```
<!DOCTYPE AuthorBook [  
  <!ELEMENT AuthorBook      (Author*, Book*)>  
  <!ELEMENT Author (Name, Address+)>  
    <!ATTLIST Author  
      AuthorId      ID          #REQUIRED  
      writing        INREFS     #IMPLIED  
    >  
  <!ELEMENT Name          (#PCDATA)>  
  <!ELEMENT Address       (Street, City)>  
  <!ELEMENT Street        (#PCDATA)>  
  <!ELEMENT City          (#PCDATA)>  
  <!ELEMENT Book          (Title, Publisher, Year)>  
    <!ATTLIST Book  
      BookIn      ID          #REQUIRED  
      authorOf    IDREFS     #REQUIRED  
    >  
  <!ELEMENT Title         (#PCDATA)>  
  <!ELEMENT Publisher     (#PCDATA)>  
  <!ELEMENT Year          (#PCDATA)>  
>
```

6. (10) According to the above DTD, construct an Xpath expression to find the author's name who published a book entitles "Art of Programming" in 1972.