Εργασία Δεύτερη
"XML Trees"

1. Γενική περιγραφή της άσκησης

Ο σκοπός της παρούσας άσκησης είναι η υλοποίηση ενός parser ο οποίος αναλύει αρχεία τύπου xml. Τα αρχεία τύπου xml έχουν προτυποποιημένη ιεραρχική δομή και ο parser δημιουργεί ένα δένδρο το οποίο απεικονίζει αυτήν την ιεραρχία για ένα συγκεκριμένο αρχείο.

2. Γλώσσα XML

Η γλώσσα XML (Extensible Mark-Up Language) είναι μία γλώσσα αναπαράστασης και περιγραφής δεδομένων. Για την περιγραφή των δεδομένων χρησιμοποιούνται tags τα οποία ορίζονται από τον ίδιο το χρήστη. Για παράδειγμα είναι δυνατόν να οριστούν tags που περιγράφουν ένα βιβλίο ως

```
<book>
   <author>G. S. Tselikis</author>
   <author>N. D. Tselikas</author>
   <title>C: From Theory to Practice</title>
   <genre>Computer Programming</genre>
   <publish_date>June 25, 2017</publish_date>
</book>
```

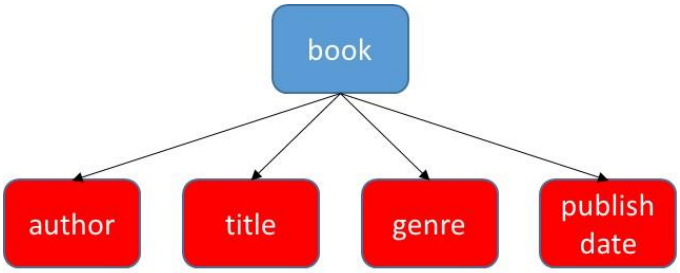
Παρατηρήστε ότι το tag **<book>** εμπεριέχει (εμφωλεύει) άλλα tags όπως **<author>**, **<title>**, **<genre>** και **<publish_date>**.

Όπως φαίνεται από το παραπάνω παράδειγμα, η σύνταξη της XML είναι αυστηρή και υπακούει στους εξής κανόνες:

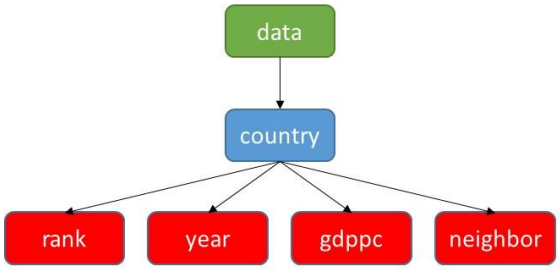
- Τα tags «ανοίγουν» **<book>** και «κλείνουν» **</book>**.
- Τα tags είναι «αυστηρά εμφωλευμένα» δηλαδή «κλείνει» πάντα πρώτο το τελευταίο tag που «άνοιξε» (**<book><author>** .... **</author></book>**).
- Τα tags είναι case sensitive.

3. Δένδρο XML

Οι παραπάνω κανόνες δίνουν τη δυνατότητα να απεικονιστεί η περιγραφή του βιβλίου με ένα δένδρο. Στο συγκεκριμένο παράδειγμα, ρίζα του δένδρου είναι το tag **<book>** και όλα τα εμφωλευμένα tags είναι παιδιά αυτής της ρίζας.



Αν τώρα υπάρχουν περισσότερα επίπεδα εμφώλευσης τότε το xml δένδρο αποκτά περισσότερα επίπεδα, για παράδειγμα:




```
<data>
  <country name="Liechtenstein">
    <rank>1</rank>
    <year>2008</year>
    <gdppc>141100</gdppc>
    <neighbor name="Austria" direction="E"/>
    <neighbor name="Switzerland" direction="W"/>
  </country>
  <country name="Singapore">
    <rank>4</rank>
    <year>2011</year>
    <gdppc>59900</gdppc>
    <neighbor name="Malaysia" direction="N"/>
  </country>
</data>
```

Συνεπώς το xml δένδρο είναι ένα *M*-αδικό δένδρο με ύψος το οποίο καθορίζεται από το «βάθος» των εμφωλευμένων tags. Το πλήθος παιδιών *M* καθορίζεται από το «πλάτος» των εμφωλευμένων tags.

4. Υλοποίηση

Για την υλοποίηση θα πρέπει να ορίσετε ένα *M*-αδικό δένδρο. Θεωρήστε ότι το πλήθος παιδιών *M* είναι σταθερό και γνωστό εκ των προτέρων.

Ο parser μπορεί να βασιστεί στη συνάρτηση parse() που δημιουργεί το δένδρο συντακτικής ανάλυσης, το οποίο διδαχθήκατε στη θεωρία. Συγκεκριμένα, η υλοποίηση μπορεί να βασιστεί στον ψευδο-κώδικα

```
tree parse(tree parseroot, char *tag){

 Εάν το υπο-δένδρο με υπο-ρίζα parseroot είναι κενό
      εισήγαγε το tag στην υπο-ρίζα;

 while(1){

      Ανίχνευσε την αρχή ή τέλος ενός tag;

      Εάν ανίχνευσες τέλος ενός tag επέστρεψε την υπο-ρίζα parseroot;

      Εάν ανίχνευσες αρχή ενός tag{

            Αν το tag βρίσκεται σε κάποιο παιδί (έστω k)
                  parseroot->child[k]=parse(parseroot->child[k],tag)

            Αλλιώς{
                  Βρες το πρώτο παιδί j με κενό υπο-δένδρο;
                  parseroot->child[j]=parse(parseroot->child[j],tag);
            }
      }
 }
}
```

Παρατηρήστε ότι ο παραπάνω ψευδοκώδικας ανιχνεύει tags και δημιουργεί υπο-δένδρα όταν τα tags δεν υπάρχουν ήδη στο δένδρο xml. Εάν τα tags υπάρχουν απλώς διασχίζει τα αντίστοιχα υποδένδρα.

Μερικές υποδείξεις για την υλοποίηση:

- Προτιμήστε να διαβάζετε μεμονωμένους χαρακτήρες από το αρχείο. Χαρακτήρες όπως οι '<', '>' και '/' θα πρέπει να έχουν ιδιαίτερη αντιμετώπιση καθώς χρησιμοποιούνται για να ξεκινήσουν και να τελειώσουν tags.
- Τα tags τυπικά ξεκινούν ως **<book>** και καταλήγουν με **</book>**.
- Εναλλακτική σύνταξη είναι και η

  <neighbor name="Malaysia" direction="N"/>

  Σε αυτή περίπτωση το tag έχει attributes, τα οποία δε θα χρησιμοποιηθούν στην άσκηση. Παρατηρήστε ότι η έναρξη είναι ελαφρώς τροποποιημένη (**<neighbor**) όπως και η λήξη (**/>**). Το πρόγραμμά σας θα πρέπει να να αντιμετωπίζει συνδυασμούς έναρξης και λήξης tags όπως

  <neighbor name="Malaysia" direction="N"> ... </ neighbor >

Ζητούμενο 1 (Βάρος 90%)

Υλοποιήστε ένα πρόγραμμα το οποίο θα κατασκευάζει το δένδρο xml από ένα xml αρχείο εισόδου. Το πρόγραμμα θα απεικονίζει το δένδρο δείχνοντας το επίπεδο που βρίσκεται το κάθε tag, για παράδειγμα:

```
<book>
      <author>
      <title>
      <genre>
      <publish_date>
```

```
<data>
    <country >
        <rank>
        <year>
        <gdppc>
        <neighbor>
```

Μπορείτε να βρείτε πλήθος αρχείων στο Διαδίκτυο για να ελέγξετε την εκτέλεση του προγράμματός σας. Χρησιμοποιήστε αρχεία τα οποία είναι συντακτικώς ορθά – δεν απαιτείται το πρόγραμμά σας να ελέγχει την ορθότητα της σύνταξης.

Ζητούμενο 2 (Βάρος 10%)

Τροποποιήστε το πρόγραμμά σας ώστε να ελέγχει ότι τα tags είναι εμφωλευμένα σωστά. *Υπόδειξη: συγκρίνετε το αλφαριθμητικό λήξης του tag με αυτό της τρέχουσας υπο-ρίζας.*

Παραδοτέα

1. Κώδικας με σχόλια και όποια εξωτερικά αρχεία χρησιμοποιήσετε. Ο κώδικας πρέπει να αναφέρει τα μέλη της ομάδας (μέχρι δύο άτομα) και να ανέβει στο e-class μέχρι την ημερομηνία υποβολής. Ο κώδικας θα πρέπει να τρέχει σωστά σε μηχάνημα του Τμήματος (π.χ. Helios, εργαστήριο Dell/Alienware).
2. Αναφορά σε έντυπη μορφή στην οποία θα πρέπει να παράγετε αποτελέσματα για ενδεικτικές εκτελέσεις. Η αναφορά δε θα ανέβει στο e-class, αλλά θα την έχετε μαζί σας κατά την εξέταση της εργασίας.

Καλή Επιτυχία