Assignment 1

File Convertor - Bin2XML

Due date: 24.04.2022

In this assignment, you should develop a command line tool to convert a binary file to XML format. All coding must be in C programming language.

This assignment expects to help you practice basic file operations and understand details of file formats, as well as practicing C programming language. In line documentation is expected, as well as good coding practices such as consistent naming, proper usage of indentation and high readability of code.

Assignment should be done as a group of 2 people.

Implementation Details & Requirements

We give a sample binary file ("*records.dat*") to test your program. This file consists of some information about the employees of a company. Each record has the following attributes and their sizes are given below:

```
struct record {
    char name[64];              //utf16
    char surname[32];           //utf8
    char gender;
    char email[32];
    char phone_number[16];
    char address[32];
    char level_of_education[8];
    unsigned int income_level;      // given little-endian
    unsigned int expenditure;    // given big-endian
    char currency_unit[16];
    char currentMood[32];
    float height;
    unsigned int weight;

};
```

The details of the attributes

Attribute Name	Description
name	Name of a person
surname	Surname of a person
gender	"*M*" or "*F*"
email	email addresses
phone_number	Phone number
address	Address of the person

level_of_education	"*PhD*", "*MSc*", "*BSc*", "*HS*", or "*PS*" where *PhD*: Doctor of Philosophy, *MSc*: Master of Science, *BSc*: Bachelor of Science,..
income_level	Income level of the person
expenditure	Expenditure of the person
currency_unit	"€", "₺", or "$"
current_mood	Emojis: "□", "☹", …
height	Height of the person
weight	Weight of the person

Your program will **produce an XML file(not include all attributes)** that looks like

```
<records>
      <row id="1">
            <name>James</name>
            <surname>Butt</surname>
            <gender>M</gender>
            <email>jbutt@gmail.com</email>
            <phone_number>504-845-1427</phone_number>
            <address>7 W Cerritos Ave #54</address>
            <level_of_education>MSc</level_of_education>
            <currency_unit>$</currency_unit>
            <height>1.33</height>
            <weight>68</weight>

      </row>
      ...
</records>
```

Please pay attention that the root element of the output XML file is the name of the output file.

For each person that was read from the file, a row number is assigned as the "id" attribute starting from 1 and its value is increased by 1. The element names of "row" in the XML file are found at the beginning of the "*records.dat*" file.

The field "*surname*" should be read in UTF-8 format whereas the field "*name*" should be read in UTF-16 format.

The field "*expenditure*" should be read in the Big Endian format whereas the field "*income_level*" should be read in the Little Endian format.

Usage of Bin2XML must be

```
Bin2XML <inputfile> <outputfile>
```

The first argument, <inputfile> refers to the source filename to be used for the conversion and the second one, <outputfile>, refers to the target XML filename.

The sample command line usage converting from the binary file to XML as follows:

```
Bin2XML personList.dat personList.xml
```

You should also create an XSD file that will be used to validate your XML. XSD file should include all necessary properties including patterns and restrictions. After the creation of the XML and XSD files, your program should automatically validate the XML file using the XSD file and return the result of validation by printing "Validation Completed." or "Validation Error."

Documentation

In this assignment, your report include

- Detail explanation of your solution
- Sample screenshots
- Problems you encounter

Submission

- Name your file *xxx*.c, *xxx*.doc, *xxx*.pdf, .. where *xxx* is your ***student id1_ student id2***
- Late submission is accepted but, 10 points penalty applies for each day.

Honesty

Your submissions will be scanned among each other as well as the Internet repository. Any assignments that are over the similarity threshold of a system for Detecting Software Similarity will get zero. We strongly encourage you to submit your assignment that has missing part rather than a dishonest submission.

Grading policy

- Binary file reading - %35
- XML file creation - %30
- UTF-8 & UTF-16 & Endian format reading - %10
- XSD validation - %15
- Document - %10

For any questions about the assignment please write under the topic “Assignment1 Questions” in Forum on the SAKAI platform. Before asking your question, please check carefully previous questions and answers, where similar questions that were asked by someone else were already answered.