Test 3

12/05/2017

*****You have 60 minutes to complete the exam in class.*****

Name ____________________________

UBIT ____________________________

Person # ____________________________

Directions. Read and understand each problem before you start to answer. Provide complete answers, show your work, and do not forget to explain and/or justify an answer when needed. Strive for brevity, clarity, and completeness in your answers. Answer on the spaces provided.

Problem 1 (65 Points) You are given an XML file ”university.xml” that is valid against the following DTD:

```
<!DOCTYPE University [
<!ELEMENT University (department+, course+, student*)>
<!ELEMENT department EMPTY>
<!ATTLIST department did ID #REQUIRED
name CDATA #REQUIRED>
<!ELEMENT course (instructor+)>
<!ATTLIST course cid ID #REQUIRED
cdept IDREF #REQUIRED>
<!ELEMENT instructor (#PCDATA)>
<!ELEMENT student (address+, courseInfo+)>
<!ATTLIST student sid ID #REQUIRED
sname CDATA #REQUIRED
sdept IDREF #REQUIRED>
<!ELEMENT address (street,city)>
<!ELEMENT street (#PCDATA)>
<!ELEMENT city (#PCDATA)>
<!ELEMENT courseInfo (grade)>
<!ATTLIST courseInfo crid IDREF #REQUIRED>
<!ELEMENT grade (#PCDATA)>
]>
```

Where attribute *cdept,sdept* reference to *did*, and *crid* references to *cid*. Use *doc("university.xml")* to access to the given database, write following queries in XPath or XQuery.

1.1 (10 points) Write a query in XPath to find the first address of the student with *sid* "s1234".

```
doc("university.xml")/University/student[@sid="s1234"]/address[1]
```

1.2 (15 points) Write a query in XQuery to return all students who have an address "234 SweetHome St. Buffalo", where "Buffalo" is the *city* and "234 SweetHome St." is the *street*, the result should be sorted in ascending order by student id(*sid*).

```
let $u:=doc("university.xml")/University
for $s in $u/student
for $a in $s/address
where $a/city = "Buffalo"
and $a/street = "234 SweetHome St."
order by $s/@sid
return $s
```

1.3 (15 points) Write a query in XQuery to return all the students who took at least one course taught by John Smith (i.e. the *instructor* of the course is John Smith).

```
let $u:=doc("university.xml")/University
let $jCourse:= (for $ins in $u/course
where $ins/instructor="John Smith"
return $ins)
let $stu:= $u/student
for $s in $stu,
    $c in $jCourse
where data($s/courseInfo/@crid) = data($c/@cid)
return $s
```

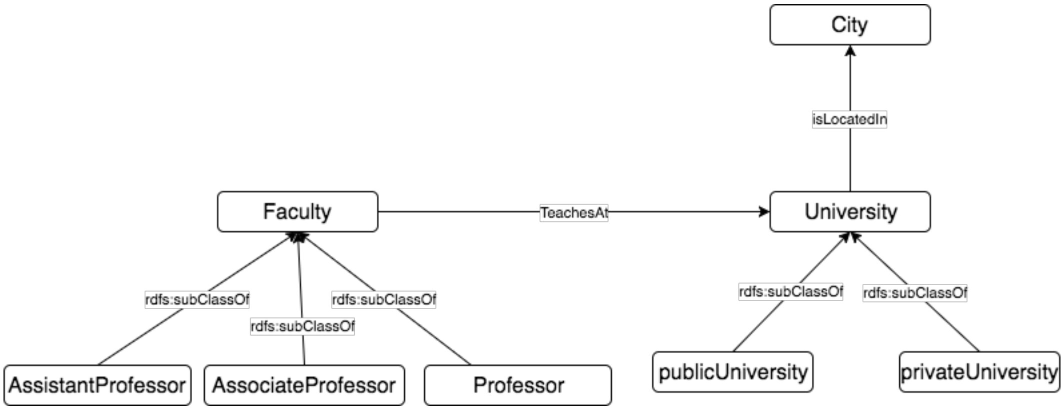
1.4 (25 points) Write a query in XQuery to return all ids(*sid*) of students who never received a *grade* less than 3.5 and have taken more than 3 courses.

```
let $u:=doc("university.xml")/University
let $sh:=(for $s in $u/student
where every $g in $s/courseInfo/grade satisfies
$g >= 3.5
return $s)

let $sa:=(for $s in $u/student
where count($s/courseInfo)>3
return $s)

for $s1 in $sh, $s2 in $sa
where data($s1/@sid) = data($s2/@sid)
return data($s1/@sid)
```

Problem 2 (25 Points) Given the following RDF schema:.



2.1(10 points) Write a query in SPARQL to find the universities located in Buffalo or Albany. You can assume the instances of the class City are labelled by the name of the city.

```
SELECT ?univ
WHERE {{?univ isLocatedIn Buffalo} UNION {?univ isLocatedIn Albany}}
```

2.2(15 points) Write a query in SPARQL to find the professors that teach at some public university in Buffalo. You can assume the instances of the class City are labelled by the name of the city.

```
SELECT ?prof
WHERE {
?prof rdf:type Professor.
?prof TeachesAt ?univ.
?univ isLocatedIn Buffalo.
?univ rdf:type publicUniversity.}
```

Problem 3 (10 points)
3.1 (5 points) Explain the differences between the operator = and **eq** in

XQuery.

Sol: See slide 47 of the XQuery slides.

3.2 (5 points) Explain the differences between Well-formed XML and Valid XML.

Sol: See slide 24 of the XMLSchema slides.