INFS 740 Notes Week 02

Q.1 All students SSN who took CS 550 and got A

[

Let $transcript : = $db.tables.transcripts []

For $ t is $transcripts

Where $t.dcode = “CS” and

$t.cno = 530 and

$t.grade = “A”

Return $t.ssn

]

Q.2 All students name and SSN who took CS 55O got A

[

Let $transcript : = $db.tables.transcripts []

Let $studnet := $db.tables.student []

For $ t is $transcript, $s in $student

Where $t.dcode = “CS” and

$t.cno = 550 and

$t.grade = “A” and

$t.ssn =$

Return {ssn : $t.ssn,

Name : $s.name

}

]

INFS 740 Notes Week 03

ATOM “prettifies”

Boolean Query 1: For every course, the number of units ≤ 6.

every $c in $course

satisfies $c.units <=6

Boolean Query 2: Everyone who is enrolled in a class must be a student.

every $e in $enrollment

satisfies (some $s in $student

satisfies $e.ssn = $s.ssn

)

satisfies must follow a Boolean expression

A → B

|  |  |  |
| --- | --- | --- |
| A/B | T | F |
| T | T | F |
| F | T | T |

Boolean Query 3: for courses with cno < 400 the possible grades are A, B or C. (Some courses may not appear in the transcript yet. Those courses have to be returned too.)

every $c in $course

satisfies if ($c.cno < 400)

then (every $t in $transcript

satisfies (if ($t.dcode = $c.dcode and $t.cno = $c.cno)

then $t.grade = “A” or

$t.grade = “B” or

$t.grade = “C”

)

else true

else true

Can simplify the query by using just transcript.

Boolean Query 3’: for some courses with cno < 400 the possible grades are A, B or C.

some $c in $course

satisfies if ($c.cno < 400)

then (every $t in $transcript

satisfies (if ($t.dcode = $c.dcode and $t.cno = $c.cno)

then $t.grade = “A” or

$t.grade = “B” or

$t.grade = “C”

)

else true

else true

In this case, we cannot simplify by just using transcript.

Boolean Query 4: The instructor for a class offered by a department must be a faculty in that department.

every $c in $class

satisfies (some $f in $faculty

satisfy $f.ssn = $c.instr

and $f.dcode = $c.dcode

)

Need to use some, not every, because otherwise …

Boolean Query 5: Students majoring in CS must be enrolled in a class titled “vocation in Hawaii”.

every $s in $student

satisfies if $major = “CS”

then (some $e in $enrollment,

$cl in $class,

$c in $course

satisfies $s.ssn = $e.ssn and

$e.class = $cl.class and

$cl.dcode = $c.dode and

$cl.cno = $c.cno and

$c.title = “vocation in Hawaii”

else true)

Boolean Query 7: Students enrolled in a class must have taken (in transcript) all prerequisite courses with the grade of B or better.

every $e in $enrollment

satisfies (every $cl in $class,

$p in $prereq

satisfies if $e.class = $cl.class and

$d.dcode = $p.dcode and

$cl.cno = $p.cno

then some $t in $transcript

satisfies ($t.ssn = $e.ssn and

$t.dcode = $p.pcode and

$t.cno = $p.pno and

($t.grade = “A” or $t.grade = “B”)

)

else true

every $e in $enrollment

satisfies mm:StudentSatClassPrereqs($e.ssn, $e.class, $univDB)

declare function mn: StudentSatClassPrereqs($ssn, $class, $univDB)

Data Query 1: Give an array of courses {dcode: …, cno: …} along with the name of students who are currently taking them.

[ {dcode: … , cno: … , students: [ {ssn: … , name: … , major: … , status: …}, …] },

…

]

FLWR generates a sequence.

Use distinct\_values to remove duplicates; works on atomic items.

[

for $c in $course

let $student\_ssns := distinct\_values (

(for $e in $enrollment,

$cl in $class

where $e.class = $cl.class and

$cl.dcode = $c.dcode and

$cl.cno = $c.cno

return $e.ssn

)

)

let $students := [ for $ssn in $student\_ssns,

$s in $student

where $s.ssn = $ssn

return $s

]

return {dcode: $c.dcode, cno: $c.cno, students: $students}

]

Data Query 4: Give students [ { ssn: … , name: … , major: … , status: …}, … ] who are currently enrolled in INFS740 and have satisfied all prereqs.

The answer to the query is an array.

[

for $s in $student

where some $e in $enrollment,

$c in $class

satisfies $s.ssn = $e.ssn and

$e.class = $c.class and

$c.dcode = “INFS” and

$c.cno = 740 and

(mm:StudentSatCoursePrereqs ($s.ssn, $c.dcode, $c.cno, $univDB) )

return $s

]