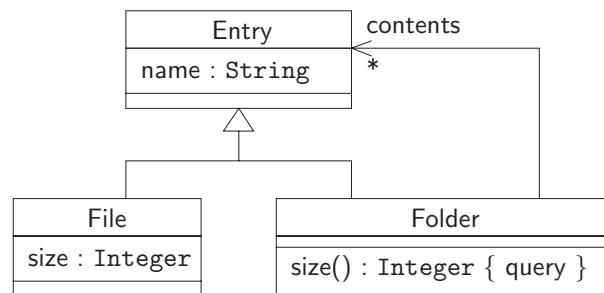


Kurzprüfung zu Formale objektorientierte Software-Entwicklung

Aufgabe 1

(10 Punkte)

Es sei das folgende Klassendiagramm als Modell für Directory-Einträge in einem Filesystem gegeben:



- Es sei $\Sigma_{\Delta}^{\text{OCL}} = (S_{\Delta}^{\text{OCL}}, \leq, OP_{\Delta}^{\text{OCL}})$ die über diesem Klassendiagramm Δ definierte OCL-Signatur. Geben Sie von S_{Δ}^{OCL} die Klassensorten sowie die daraus abgeleiteten Kollektionssorten sowie deren Untersortenbeziehungen in \leq an. Geben Sie weiter von OP_{Δ}^{OCL} die aus Attributen, Assoziationsenden und Queries abgeleiteten Operationen an.
- Entscheiden Sie, ob die im Folgenden aufgeführten Zeichenketten korrekte OCL-Ausdrücke über diesem Klassendiagramm und der Variablenmenge $\{g : \text{Folder}, e : \text{Entry}, s : \text{Integer}\}$ darstellen. Geben Sie für die korrekten OCL-Ausdrücke den kleinsten Typ an. Geben Sie in den restlichen Fällen eine kurze Begründung an, warum die Zeichenketten keinen korrekten OCL-Ausdruck darstellen.
 - `Folder.allInstances()->union(File.allInstances())`
 - `g.contents->iterate(e : Entry; s : Integer = 0 | if e.oclIsTypeOf(File) then s+e.size else s+e.size() endif)`
 - `g.contents.size() > 0 and g.size()/g.contents.size() < 0`
 - `g.contents@pre->exists(e : Entry | e <> null)`
 - `g.contents->select(e : Entry | e.oclIsTypeOf(File))`
 - `g.size() = g`

Aufgabe 2

(2 Punkte)

Geben Sie die semantische Funktion für die strikte OCL-Multiplikation

$$\llbracket _*__ \rrbracket : \llbracket \text{Integer} \rrbracket \times \llbracket \text{Integer} \rrbracket \rightarrow \llbracket \text{Integer} \rrbracket$$

an.

Aufgabe 3

(4 Punkte)

Zeigen Sie, dass für alle $x, y \in \llbracket \text{Boolean} \rrbracket$ gilt:

$$\llbracket _or__ \rrbracket(x, y) = \llbracket not \rrbracket(\llbracket _and__ \rrbracket(\llbracket not \rrbracket(x), \llbracket not \rrbracket(y)))$$