

Laboratorul 7: MicroHaskell și Model Test

MicroHaskell

Fișierul `microHaskell.hs` conține un mini-limbaj funcțional, împreună cu semantica lui denotațională, așa cum a fost definit în **cursul 6**. Definim comanda:

```
run :: Hask -> String
run pg = showV (hEval pg [])
```

Astfel, `run pgm` va întoarce rezultatul evaluării (rulării) programului `pgm`.

- 4.1) Scrieți mai multe programe și rulați-le pentru a vă familiariza cu sintaxa.
- 4.2) Adăugați operația de înmulțire pentru expresii, cu precedență mai mare decât a operației de adunare. Definiți semantica operației de înmulțire.
- 4.3) Folosind funcția `error`, înlocuiți acolo unde este posibil valoarea `VError` cu o eroare care să precizeze motivul apariției erorii.
- 4.4) Adăugați expresia `HLet Name Hask Hask` ca alternativă în definirea tipului `Hask`. Semantica acestei expresii este cea uzuală: `HLet x ex e` va evalua `e` într-un mediu în care `x` are valoarea lui `ex` în mediul curent. De exemplu, dacă definim

```
h1 = HLet "x" (HLit 3) ((HLit 4) :+: HVar "x")
```

atunci `run h1` va întoarce `"7"`.

Model Test

Deschideți fișierul `modeltest.pdf` și lucrați exercițiile din acesta.