

## Documentatie laborator 5 – Inchideri functionale. Fluxuri implementate cu inchideri functionale.

### Inchideri functionale (closures)

Cititi slide-urile 28-32 din cursul 4. Un aspect remarcabil al inchiderilor functionale este ca ele pot fi folosite pentru a intarzia evaluarea. Cum asa? O inchidere functionala este compusa dintr-o functie si un context. Cand exploatham faptul ca functiile sunt valori de ordinul 1 (le pasam ca argumente, le returnam, le folosim pe post de componente in structuri mai mari etc), valoarea cu care umblam pe post de functie este retinuta ca un text si un context, fara a se incerca evaluarea corpului functiei. De exemplu, cand lucram cu functia  $(\lambda (x) (+ a x))$ , nu incercam sa il adunam pe a cu x, ci doar retinem textul functiei si valoarea lui a la momentul respectiv. Adunarea se va produce numai atunci cand aplicam efectiv functia pe un argument. Consecinta este ca putem inchide orice calcul pe care vrem sa il amanam pe mai tarziu intr-o functie ( $\lambda ()$  calcul) si sa provocam evaluarea la momentul dorit aplicand aceasta functie (aici pe 0 argumente). Amintiti-va ca am facut acest lucru la cursul 3 pentru a defini if-0 ca pe o functie nestricta (v. curs3-ex.rkt).

Pentru a intelege mai bine cum functioneaza inchiderile functionale, experimentati cu exercitiul 1 din lab5-doc.rkt.

### Fluxuri implementate cu inchideri functionale

Pana acum am lucrat intensiv cu liste si cu functionale capabile sa manipuleze liste in cele mai uzuale situatii. Ce se intampla insa cand vrem sa lucram cu insiruirii oricat de lungi de elemente, de exemplu cu lista tuturor numerelor naturale? Nu putem memora toate numerele naturale intr-o lista pentru a le manipula ulterior. Sau ce se intampla daca vrem sa prelucram in diferite feluri o lista de foarte multe elemente? E oare cea mai buna solutie sa pasam de colo-colo o structura de date care ocupa atata memorie? Oare nu se va intampla de multe ori sa nu avem cu adevarat nevoie de toate elementele listei?

O solutie pentru aceasta problema sunt fluxurile – liste cu evaluare intarziata. Fluxurile exploateaza ideea inchiderilor functionale, definindu-se ca o pereche dintre primul element si, intr-o inchidere functionala, restul fluxului. Inchiderea functionala actioneaza ca un motor care, cand este pornit (prin aplicatie de functie), produce o noua pereche intre un prim element si, intr-o inchidere functionala, noul rest al fluxului. Astfel cerem dintr-un flux intotdeauna exact cate elemente sunt necesare.

O implementare si un exemplu gasiti in exercitiile 2-3 din lab5-doc.rkt.