

TÖL304G

Forritunarmál

Verkefnablað 4

Snorri Agnarsson

15. september 2019

Verkefni

Þið skuluð prófa öll verkefnin ykkar í einhverju Scheme kerfi, til dæmis DrRacket og sýna niðurstöður.

Hér eru nokkur Scheme föll og lykilorð sem mögulegt er að þið viljið nota til að leysa þessi verkefni: `define`, `lambda`, `if`, `and`, `or`, `car`, `cdr`, `cons`, `null?`, `list`, `=`, `*`, `+`.

Athugið að Scheme report¹ inniheldur lýsingar á öllum þessum föllum og lykilorðum og einnig má finna nákvæma skjölun í fylgigögnum fyrir DrRacket og önnur Scheme kerfi.

Athugið að halaendurkvæmni má vera þannig að viðkomandi fall noti hjálparfall til að leysa verkefnið og að hjálparfallið sé halaendurkvæmt.

Hópverkefni

1. Skriðið halaendurkvæmt Scheme fall `revindex` sem uppfyllir eftirfarandi lýsingu:

```
;; Notkun: (revindex n)
;; Fyrir:  n er heiltala, n>=0
;; Gildi:  listi allra heiltalna i, þannig að
;;         0<i<=n, í minnkandi röð.
```

Til dæmis skal Scheme segðið `(revindex 0)` skila `()` og Scheme segðin `(revindex 5)` skal skila `(5 4 3 2 1)`. Sýnið prófanir, eins og endranær.

¹<http://www.hi.is/~snorri/downloads/r5rs.pdf>

2. Skrifðu *halaendurkvæmt* Scheme fall *foldr* sem uppfyllir eftirfarandi lýsingu:

```
;; Notkun: (foldr f x u)
;; Fyrir:  f er tvíundarfall, þ.e. fall
;;         sem tekur tvö viðföng af einhverju
;;         tagi, x er listi (x1 ... xN)
;;         gilda af því tagi, u er gildi
;;         af því tagi.
;; Gildi:  (f x1 (f x2 (f ... (f xN u) ...)))
;; Aths.: Með öðrum orðum, ef við skilgreinum
;;         tvíundaraðgerð ! með  $a!b = (f\ a\ b)$ ,
;;         þá er útkoman úr fallinu gildið á
;;          $x1\ !\ x2\ |\ \dots\ !\ xN\ !\ u$ 
;;         þar sem reiknað er frá hægri til
;;         vinstri
```

Til dæmis skal segðin `(foldr + (list 1 2) 3)` skila tölunni 6, þ.e. $1 + 2 + 3$, reiknað frá hægri. Athugið að það er nauðsynlegt að tryggja að röð reiknaðgerða sé nákvæmlega rétt þannig að, til dæmis, segðin `(foldr - (list 1 2) 3)` skili réttu gildi samkvæmt lýsingunni að ofan, þ.e. útkomunni úr segðinni `(- 1 (- 2 3))`, sem er 2. Þið munið þurfa hjálparfall, sem þið skuluð skrifa skilmerkilega lýsinu á (þ.e. Notkun/Fyrir/-Gildi). Þið megið nota innbyggða fallið `reverse`, sem snýr við lista. Til hliðsjónar er gagnlegt að kíkja á Dafny fallið `FoldR_loop` sem er hér² á vefnum.

Sýnið prófun á segðunum `(foldr - '(2 3) 1)` og `(foldr cons '(1 2) '())` og `(foldr cons '() 1)`.

Athugið að rökstuðningur hjálparfallsins er lykilatriði í þessu dæmi. Ekki fást mörg stig fyrir dæmið ef rökstuðningurinn er gallaður.

3. Notið föllin að ofan (`revindex` og `foldr`) til að skrifa tvær Scheme segðir til að reikna summu og margfeldi talnanna $1, \dots, 20$.

Einstaklingsverkefni

1. Skrifðu Scheme fall `foldrec` sem hefur sömu lýsingu og `foldr` að ofan, en skal ekki vera *halaendurkvæmt*. Ekki má skrifa nein hjálparföll. Þetta ætti að vera létt verk fyrir þá sem skilja grundvallaratriði í Scheme.

Sýnið prófun á segðunum `(foldrec - '(2 3) 1)` og `(foldrec cons '(1 2) '())` og `(foldrec cons '() 1)`.

²<https://rise4fun.com/Dafny/xR7n>

2. Skrifðu halaendurkvæmt Scheme fall `foldl` sem uppfyllir eftirfarandi lýsingu:

```
;; Notkun: (foldl f u x)
;; Fyrir:  f er tvíundarfall, þ.e. fall
;;         sem tekur tvö viðföng af einhverju
;;         tagi, x er listi (x1 ... xN)
;;         gilda af því tagi, u er gildi
;;         af því tagi.
;; Gildi:  (f ... (f (f u x1) x2) ... xN)
;; Aths.: Með öðrum orðum, ef við skilgreinum
;;         tvíundaraðgerð ! með  $a!b = (f\ a\ b)$ ,
;;         þá er útkoman úr fallinu gildið á
;;          $u\ !\ x1\ !\ x2\ | \dots\ !\ xN$ 
;;         þar sem reiknað er frá vinstri til
;;         hægri
```

Til hliðsjónar er gagnlegt að kíkja á Dafny fallið `FoldL_loop` sem er hér³ á vefnum. Munið að halaendurkvæmt fall er hliðstætt lykkju.

Ekki má skrifa nein hjálparföll, enda eru þau algerlega ónauðsynleg í þessu verkefni.

Sýnið prófun á segðunum `(foldl - 1 '(2 3))` og `(foldl (lambda (x y) (cons y x)) '() '(1 2 3))`.

³<https://rise4fun.com/Dafny/xR7n>