Informatica Instituut, Faculteit Wiskunde en Informatica, UU.
In elektronische vorm beschikbaar gemaakt door de TEX van A−Eskwadraat.
Het college INFOFP werd in 2003/2004 gegeven door Johan Jeuring.

Functioneel Programmeren (INFOFP)
17 maart 2004

Opgave 1

Wat is de waarde van de onderstaande expressie?
`[ y-x | x <- [1,2], y <- [3..5]]` *(5 punten)*

a) `[-2,-3,-4,-1,-2,-3]`

b) `[2,3,4,1,2,3]`

c) `[[-2,-1,-3],[-2,-4,-3]]`

d) `[[2,1],[3,2],[4,3]]`

Opgave 2

Wat is de waarde van de onderstaande expressie?
`sum (takeWhile (<100) [10,50..])` *(5 punten)*

a) 60

b) 160

c) 150

d) geen antwoord, oneindige lus

Opgave 3

Gegeven is het datatype:

```
data MyList a = Cons (MyList a) a | Nil
```

Wat is het type van `Cons 'a' Nil`? *(5 punten)*

a) `MyList`

b) `MyList Char`

c) `MyList a -> a -> MyList a`

d) typeringsfout

Opgave 4

Schrijf het meest algemene type op van de onderstaande expressie. Een afleiding is niet nodig. Als de expressie een typeringsfout bevat, schrijf dan op: 'typeringsfout'.

`foldr (++)` *(10 punten)*

Opgave 5

Gegeven is het volgende datatype:

```
data Boom a = Tak (Boom a) (Boom a) | Blad a
```

Wat is het type van de constructorfunctie `Tak`? *(5 punten)*

Opgave 6

Schrijf een functie die telt hoeveel klinkers (a, e, i, o en u) er voorkomen in een `Boom` die letters (`Char`) bevat. Noem de functie `telKlinkers` en schrijf ook het type op. *(15 punten)*

Opgave 7

Schrijf een functie die bepaalt of alle elementen van een `Boom` voldoen aan een gegeven predicaat. Noem deze functie `allBoom` en schrijf ook het meest algemene type op. Het predicaat is een functie die gegeven een element een `Bool` oplevert. Voorbeeld van gebruik:
`allBoom even (Tak (Blad 2) (Tak (Blad 3) (Blad 4)))` geeft `false`. *(15 punten)*

Opgave 8

Breid het datatype `Boom` uit, zodanig dat ook lege bomen gerepresenteerd kunnen worden.
(10 punten)

Opgave 9

Schrijf een functie `filterBoom`, die, gegeven een predicaat en een `Boom`, alle bladeren die niet voldoen aan het predicaat vervangt door de lege `Boom`. Takken en bladeren die wel aan het predicaat voldoen worden ongemoeid gelaten. Schrijf ook het meest algemene type op.
(15 punten)

Opgave 10

Schrijf een functie die **alle** lege bomen verwijdert uit een niet-lege boom:

```
verwijderLeeg ::  Boom a -> Boom a
```

Hierbij nemen we aan dat voor alle bomen geldt dat een `Tak l r` met een lege deelboom (`l` of `r`) gelijk is aan de andere deelboom (`r` of `l`). *(15 punten)*