

# Studie van Declaratieve Talen

## Haskell 1

### Gequoteerde Oefening

#### Richtlijnen

Los onderstaande opgave zelfstandig op (samenwerking is **niet** toegestaan!), en stuur ze binnen de 24 uren na de zitting in.

Zorg ervoor dat je programma correct is. Indien dit niet zo is, moet er in commentaar bijstaan wat er niet werkt. Zorg er voor dat je programma ook strikt de specificatie volgt zoals gevraagd (bvb. niet afwijken van de opgegeven functienamen). Schrijf je oplossing in 1 enkel bestand dat je `oplossing.hs` noemt. Voeg vooraan het bestand de volgende lijnen toe:

```
% Voornaam Naam  
% Richting
```

Waarbij je jouw eigen gegevens invult. Als je oplossing klaar is, controleer dan eerst of de functienamen overeenstemmen met de opgave. Stuur ze vervolgens via Toledo in (onderdeel Toetsen en opdrachten). Opgelet: je kan slechts één keer een oplossing indienen. **Vergeet niet op Submit te klikken om je oplossing effectief door te sturen.**

Indien je vragen hebt over deze opgave kan je mailen naar

`anton.dries@cs.kuleuven.be`.

## Opgave

De driehoek van Pascal is een voorstelling die vaak gebruikt wordt om binomiaalcoëfficiënten voor te stellen. Het is zeer eenvoudig om deze driehoek rij per rij op te bouwen aan de hand van de volgende twee observaties:

- Elke rij begint en eindigt met een 1.
- De overige getallen in de rij kunnen bekomen worden door telkens twee opeenvolgende elementen uit de vorige rij op te tellen.

Dit principe is geïllustreerd in de figuur op <http://en.wikipedia.org/wiki/File:PascalTriangleAnimated2.gif>.

**Opgave** Schrijf de functie `pascal` die de driehoek van Pascal berekent *volgens de bovenstaande methode*. Deze functie heeft als resultaat een lijst waarvan ieder element een rij in de driehoek voorstelt. **Hint:** Het kan handig zijn om eerst een hulpfunctie te schrijven die een (deel van een) rij in de driehoek genereert op basis van de gegeven vorige rij.

Je kan de bestaande functie `take` gebruiken om een deel van de driehoek op te vragen, bijvoorbeeld

```
Main> take 10 pascal
[[1],
 [1,1],
 [1,2,1],
 [1,3,3,1],
 [1,4,6,4,1],
 [1,5,10,10,5,1],
 [1,6,15,20,15,6,1],
 [1,7,21,35,35,21,7,1],
 [1,8,28,56,70,56,28,8,1],
 [1,9,36,84,126,126,84,36,9,1]]
```

geeft de eerste 10 elementen uit het resultaat terug. (De line-breaks in het resultaat zijn manueel toegevoegd.)

**Schrijf ook typedeclaraties voor alle functies die je definieert!**

**Veel succes!**