WISKUNNEND WISKE

FABULOUS FIBONACCI FUN

De Fibonacci getallen zijn ongetwijfeld een van de beroemdste uit de wiskunde. Het 0^de^ Fibonacci-getal is 0 en het 1^ste^ Fibonacci-getal is 1 en voor alle natuurlijke getallen $n > 1$ is het n^{de} Fibonacci-getal gelijk aan de som van de 2 voorgaande Fibonacci getallen. De eerste zeven Fibonacci-getallen zijn dus 0, 1, 1, 2, 3, 5, en 8. De manier waarop de rij van Fibonacci wiskundig is gedefinieerd, is een voorbeeld van een recursieve definitie:

$\mathrm{Fib}(0) = 0$
$\mathrm{Fib}(1) = 1$
$\mathrm{Fib}(n) = \mathrm{Fib}(n-1) + \mathrm{Fib}(n-2)$, voor $n > 1$

De functie Fib wordt gedefinieerd in functie van zichzelf. We kunnen de functie Fib dan als volgt neerschrijven in een programmeertaal:

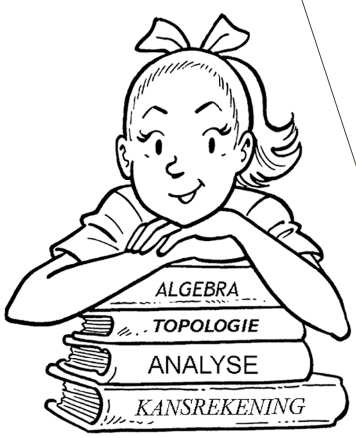
Invoer: *n*, een natuurlijk getal waarvoor we het Fibonacci-getal willen berekenen
Uitvoer: het n^{de} Fibonacci-getal

Code:	Toelichting:
Fib(*n*) {	Fib(*n*) berekenen we als volgt:

if (*n* = 0) {
        return 0
    }
	als *n* gelijk is aan 0,
	geef 0 als uitvoer

else if (*n* = 1) {
        return 1
    }
	als *n* gelijk is aan 1,
	geef 1 als uitvoer

else {
        return Fib(*n* – 1) + Fib(*n* – 2)
    }
}
	in alle andere gevallen,
	roep Fib(*n* – 1) en Fib(*n* – 2) op
	en geef hun som als uitvoer



Vrije Universiteit Brussel

WISKUNNEND WISKE

FABULOUS FIBONACCI FUN

FINALE OPDRACHT 4:

A) Hoe vaak roept de functie Fib zichzelf op nadat we Fib(5) oproepen?

Wiske heeft ook een andere manier gevonden om Fibonacci-getallen te berekenen, namelijk de volgende (recursieve) functie BetereFib:

Invoer: *n*, een natuurlijk getal waarvoor we het Fibonacci-getal willen berekenen
a, een natuurlijk getal dat initieel 0 is
b, een natuurlijk getal dat initieel 1 is
i, een natuurlijk getal dat initieel 0 is
Uitvoer: het *n*-de Fibonacci-getal

Code:

```
BetereFib(n, a, b, i) {

        if (i = n) {
                return a
        }
        else {
                return BetereFib(n, b, a+b, i+1)
        }
}
```

Toelichting:

BetereFib(*n*, *a*, *b*, *i*) berekenen we als volgt:

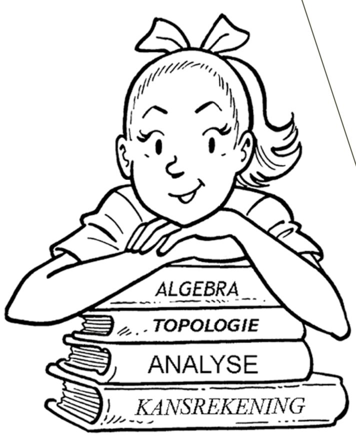
als *i* gelijk is aan *n*,
geef *a* als uitvoer

in alle andere gevallen,
roep BetereFib(*n*, *b*, *a*+*b*, *i*+1) op
en geef die waarde als uitvoer

B) Hoe vaak roept de functie BetereFib zichzelf op nadat we BetereFib(5, 0, 1, 0) oproepen?

Wiske heeft echter niet zo veel kaas gegeten van wiskundige bewijstechnieken: ze kan wat hulp gebruiken om aan te tonen dat de functies Fib(*n*) en BetereFib(*n*, 0, 1, 0) effectief hetzelfde resultaat berekenen.

C) BEWIJS dat de functie BetereFib(*n*, 0, 1, 0) wel degelijk het *n*-de Fibonacci-getal oplevert.



Je krijgt 20 minuten om deze opdracht op te lossen.
Na afloop van de voorziene tijd geeft de klasverantwoordelijke het antwoord aan de juryleden.

Vrije Universiteit Brussel