

# Zoekbomen

Kristof Teelen

Labo beeldverwerking / 2007-08

## 1 Eerste voorbeeld: Raadsel

Zoek de oplossing van het onderstaande raadsel:

U2 geeft een concert dat over 17 minuten begint, maar ze moeten eerst alle vier een brug oversteken om er te geraken. In het begin staan ze alle vier aan dezelfde kant van de brug. Het is nacht en ze hebben maar één zaklamp. Er kunnen maximaal twee mensen tegelijk oversteken. Of er nu één, of twee oversteken, de zaklamp moet altijd mee. De zaklamp moet dus steeds heen en weer meegenomen worden. Sommige bandleden worden al wat ouder en hebben dus een langere tijd nodig om de brug over te steken. Als er twee mensen tegelijk wandelen, bepaalt de langzaamste de tijd nodig om de brug over te steken: Bono heeft 1 minuut nodig om over te steken, Edge 2, Adam 5 en Larry 10 minuten. Als Bono en Larry als eersten samen oversteken, zijn er 10 minuten verstreken als ze aan de overkant aankomen. Als Larry dan terugkeert, met de zaklamp, zijn er in totaal 20 minuten verstreken, en zijn ze dus te laat voor het concert. Er is geen trukje voor de oplossing zoals zwemmen of gooien met de zaklamp. De vraag is natuurlijk: hoe doen ze het?



Figuur 1: How did U2 get over the brooklyn bridge?

Je kan eerst zo de oplossing voor het raadsel proberen te bedenken, voor je naar de code kijkt die de oplossing berekend door het opstellen van een zoekboom. Bekijk zelf ook de code in *raadsel.m*. Het is belangrijk om allereerst een goede representatie voor de data te kiezen want die bepaalt het verdere verloop en de complexiteit van het algoritme. Zou jij een andere representatie kiezen? Welke zoekmethode wordt er in de code gebruikt, breadth of depth first? Als je wil, kan je proberen om de code aan te passen zodat de andere methode gebruikt wordt.

## 2 Sudoku

Een ander raadsel dat je kan oplossen door zoekalgoritmes is de sudoku-puzzel. De spelregels zijn simpel: in elke rij, elke kolom en elk deel van rooster moeten alle getallen tussen 1 en 9 ingevuld worden. Vul de gegeven code aan met controlemechanismen zodat in elke stap van het algoritme een nieuwe, correcte knoop

9			6		4	7		
4	5				3			1
1			8	5			6	
			3		7		4	
8		7		9	2			
		2			5	3		
2						5		6
	6				8			
	1	5	9			8		7

Figuur 2: Voorbeeld sudokupuzzel

wordt aangemaakt. Voor meer informatie over de implementatie kan je ook de mathematica-code uit de theorie bekijken.