

WISKUNNEND WISKE

DE MEESTERLIJKE MACHINE



OPGAVE 2

Lara werkt in de studio die Suske & Wiske-stripverhalen tekent. Door een aardbeving is er grote schade ontstaan in het archief waarin de originele handgetekende pagina's van alle strips worden bewaard. De pagina's zelf zijn nog intact, maar liggen volledig willekeurig verspreid over de vloer. Lara staat nu voor de enorme klus om deze pagina's opnieuw te bundelen tot volledige en correcte strips, en om de stripverhalen onderling zelf ook op volgorde van nummer te krijgen.

Gelukkig staat op elke pagina achtereenvolgens het nummer van de strip (3 cijfers) en het nummer van de pagina (2 cijfers) gecodeerd. Bovendien bezit de studio een machine die pagina's kan sorteren op één van de 5 gecodeerde cijfers: wanneer je de machine bv. instelt op cijfer 4, gaat deze haar invoer sorteren op het 4de gecodeerde cijfer (dus het 1ste cijfer van het paginanummer). Verder heeft de sorteermachine volgende eigenschap: indien twee pagina's op de ingestelde plaats hetzelfde cijfer hebben, dan behoudt de machine hun onderlinge volgorde.

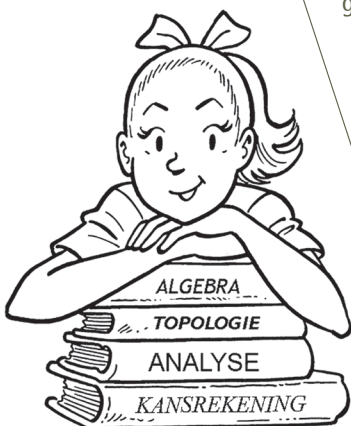
- Help Lara met het sorteren van alle pagina's op stripnummer en paginanummer: beschrijf zo duidelijk mogelijk hoe zij met behulp van de machine haar taak kan vervullen. Schrijf een stappenplan uit zodat deze werkwijze ondubbelzinnig kan uitgevoerd worden. Het stappenplan moet dus een willekeurige opeenvolging van pagina's kunnen op orde zetten door gebruik te maken van de machine. Beschrijf vervolgens ook op een ondubbelzinnige manier de interne werking van de machine.
- Hoeveel keer wordt een pagina door de machine 'vastgenomen' (= aantal operaties) wanneer je je stappenplan uit vraag a toepast op 332 strips met elk 88 pagina's? Hoeveel operaties zijn er nodig voor n strips met elk p pagina's?
- Bewijs waarom deze aanpak een willekeurige opeenvolging van pagina's in de juiste volgorde zal krijgen.

WISKUNDIG WEETJE

Binnen de computerwetenschappen is het belangrijk om op één of andere manier de **performantie van algoritmen** te kunnen vergelijken, bv. om te bepalen welk algoritme voor een gegeven probleem het snelst tot een oplossing zal leiden. Je zou de performantie kunnen meten door het algoritme in een programmeertaal uit te schrijven, en vervolgens de uitvoeringstijd van het programma te meten voor een aantal mogelijke invoerwaarden. Dit heeft echter als nadeel dat zo'n metingen afhangen van de specifieke programmeertaal waarin het algoritme is geprogrammeerd, en van de specifieke computer waarop de metingen worden uitgevoerd.

Een algemenere manier om de performantie van algoritmen te bepalen is om het algoritme zélf te gaan onderzoeken, en te bepalen hoeveel **elementaire computationele stappen** het algoritme zal vereisen om een probleem van een gegeven grootte op te lossen. Bijvoorbeeld: hoeveel stappen vereist een sorteeralgoritme om een rij van n elementen te sorteren. Zo kan men de performantie van een algoritme beschrijven door middel van een functie in één veranderlijke, nl. de grootte van het probleem. En die functies kan je dan gebruiken om voor een gegeven probleem een keuze te maken tussen verschillende algoritmen.

Performantie van algoritmen wordt in de Bachelor in de Computerwetenschappen behandeld in het vak 'Algoritmen en datastructuren 1'. In de andere Bacheloropleidingen in de Wetenschappen komt het onderwerp aan bod in het vak 'Inleiding tot de computerwetenschappen'.



Vrije
Universiteit
Brussel