

B FOR YOU

Snelle computers rekenen slim - 8

Prijzen uitgerekte voor profielwerkstukken - 6

Een loopbaan met zijpaden: bètastudie is geen fuik - 10

Dit artikel schrijf ik met mijn fancy computer, die behalve tekst verwerken nog veel meer doet: muziek spelen, de nieuwste software downloaden en mijn harde schijf op virussen controleren. Dit kan ik allemaal tegelijkertijd, omdat ik een zeer snel apparaat heb. Dat zegt echter niet alles. Snelheid hangt namelijk ook nauw samen met de manier waarop programma's geschreven zijn: 'slim' of 'niet zo slim'.

Door Matthijs Mekking

Snelle

Hoe snel je computer is, wordt bepaald door de processor en het werkgeheugen. Vooral de snelheid van de processor is belangrijk. Die noemen we kloksnelheid en we drukken deze uit in MegaHertz (MHz). Hoe sneller je processor, hoe sneller je computer gegevens verwerkt.

Het andere belangrijke onderdeel is dus het werkgeheugen, ook bekend als RAM, Random Access Memory. Dit heeft invloed op de snelheid van starten en verwerken van opdrachten. Meer werkgeheugen betekent verder, dat je makkelijker met verschillende programma's tegelijkertijd aan de gang kunt.

Maar er is nog een externe factor die meespeelt bij de snelheid, namelijk de programma's waarmee de computer moet werken. Een programma is opgebouwd uit procedures. Een algoritme omschrijft die procedures. Het laat zien wat er moet gebeuren om tot het

gewenste resultaat te komen. Die omschrijvingen kunnen ingewikkeld zijn, dan duurt het lang om een programma uit te voeren en heeft de computer meer tijd nodig om alles te verwerken.

Als het algoritme 'slim' is, zullen de processor en het geheugen minder intensief gebruikt worden. Wanneer we van een complex algoritme een slimmere kunnen maken, winnen we processortijd en geheugen, zodat er meer programma's in dezelfde tijd kunnen draaien.

SCHATTEN We willen kijken of we de tijdsduur kunnen verkleinen. Dit noemen we complexiteits-berekening: we maken een schatting van de looptijd van het programma. Hoe bepaal je hoeveel tijd er nodig is voor een programma? Daarvoor moet je weten wat er in gaat, wat de invoer is. Dit kan een film zijn die je wilt afspelen, of gewoon een getal. Uiteindelijk wordt de invoer gecodeerd als een rijtje bits, en hoe meer bits je nodig hebt, hoe meer tijd de machi-

ne nodig heeft om die invoer te verwerken. Dat coderen gebeurt door de computer zelf. Alle informatie (bits) die de computer ontvangt, wordt omgezet naar nullen en enen. Of het nu gaat om een foto, een formule met symbolen of een getal. Door het aantal uit te voeren berekeningen te tellen, kun je de looptijd schatten. We gebruiken wiskunde om deze schattingen met elkaar te vergelijken dit noemen we 'de taal der asymptoten'. Doordat we het aantal bits in de vergelijking stoppen, kunnen we verschillende complexiteiten met elkaar vergelijken.

We kunnen een programma zo veranderen, dat het sneller uitvoerbaar is. We kunnen de nieuwe versie vervolgens met de oude vergelijken om te zien of de snelheid inderdaad verbeterd is. Zo kun je bijvoorbeeld kijken of een functie sneller 'groeit' dan de ander: oftewel wat de complexiteit ervan is.



DE SNELHEID VAN JE COMPUTER HANGT NAUW SAMEN MET DE MANIER WAAROP PROGRAMMA'S GESCHREVEN ZIJN: 'SLIM' OF 'NIET ZO SLIM'.

computers?

De taal der asymptoten:

We hebben twee functies: $f(x)$ en $g(x)$.

* $f(x) = o(g(x))$ betekent dat f langzamer groeit dan g .

$$\lim_{x \rightarrow \infty} f(x) / g(x) = 0$$

* $f(x) = O(g(x))$ betekent dat f langzamer of gelijk groeit aan g . f groeit beslist niet sneller.

($x \rightarrow \infty$) er zijn constanten C , x_0 zodat voor alle $x > x_0$ geldt:

$$|f(x)| < Cg(x)$$

* $f(x) = \Omega(g(x))$ betekent dat f niet langzamer groeit dan g . f groeit dus even snel of sneller.

$$\neg (\lim_{x \rightarrow \infty} f(x) / g(x) = 0)$$

* $f(x) = \Theta(g(x))$ betekent dat f en g even snel groeien.

er zijn constanten $c_1 \neq 0$, $c_2 \neq 0$, x_0 zodat voor alle $x > x_0$ geldt:

$$c_1 g(x) < f(x) < c_2 g(x)$$

* $f(x) \sim (g(x))$ betekent dat f en g even snel groeien en bovendien van dezelfde grootte zijn.

$$\lim_{x \rightarrow \infty} f(x) / g(x) = 1$$

De ontwikkeling op het gebied van technologie gaat snel. Als jouw computer een programma niet snel genoeg kan uitvoeren, zou je een nieuwe kunnen kopen die misschien duizend keer sneller is. Dat kun je natuurlijk doen, maar er zijn ingewikkel-

de programma's die een grote invoer langzaam verwerken. Hun looptijd groeit zo snel dat ze het op je nieuwe computer niet veel beter doen. Bij deze programma's is het waard om de theorie er op los te laten en te kijken of hun complexiteit te verbeteren is.

Als dit niet kan, betekent het dat het probleem niet sneller oplosbaar is en dat je nieuwe snelle computer eigenlijk niet zoveel sneller is dan de oude.

Computers moeten ook gewoon alles uitrekenen, net als wij. Ze zijn wel veel sneller, maar als je met een moeilijk vraagstuk aankomt, dan zal de machine voldoende tijd vragen om de berekeningen uit te voeren. Daarom is het belangrijk dat programma's uit efficiënte algoritmes bestaan, die weinig van de processor vergen en weinig geheugen in beslag nemen. Hierin ligt een dankbare taak voor computerwiskundigen: een nieuw vakgebied, ontstaan doordat computers steeds ingewikkelder problemen voor hun kiezen krijgen.