

## Opa vertelt...

In de serie 'Opa vertelt' een herinnering aan de lang verlopen tijd dat computers kolossale machines waren waar gebruikers, met de nodige eerbied, pakjes ponskaarten mochten aanleveren. Ze werden nog net niet door een stoommachine aangedreven, maar dat was het dan ook.

Iedereen maakte zijn eigen programma's. Standaardprogramma's bestonden nauwelijks, als je het trof gaf een collega je toestemming een door hem gemaakt programma te gebruiken. De meest gebruikte programmeertaal was Fortran en van virtueel geheugen en interactief werken was nog geen sprake.

Een beetje slimme programmeur maakte veel gebruik van zogenaamde subroutines. Op die manier hoefde je bijvoorbeeld niet op meerdere plekken in je programma de code voor het berekenen van een standaarddeviatie te herhalen. Je kon volstaan met een 'call' naar de subroutine die deze berekening uitvoerde en je het antwoord terugstuurde.

Een groot probleem in die beginjaren was de reken nauwkeurigheid. Een computer slaat getallen maar met een beperkte precisie op, alles voorbij een vastgesteld aantal decimalen wordt afgekapt. Nu komen in veel statistische formules zaken voor als het verschil van de som van de kwadraten van de gegevens en het kwadraat van de som van de gegevens. Dat kunnen enorme getallen worden waarvan de laatste decimalen worden afgekapt. Maar daarin zit soms het verschil dat we willen berekenen en dat wordt dan nul. Je kunt kan dit zelf gemakkelijk demonstreren met een eenvoudige rekenmachine. De getallen 1, 2, 3 hebben dezelfde standaarddeviatie als 11, 12, 13 of 101, 102, 103 of 10001, 10002, 1003 etc. Probeer het maar, en maak het aantal nullen steeds groter. Afhankelijk van de rekenmachine komt er dan op een gegeven ogenblik een foutmelding. Ik heb begin jaren '80 een artikel gepubliceerd in *Kwantitatieve Methoden* waarin ik de rekennauwkeurigheid van de statistische programma's op alle Nederlandse universitaire reken centra heb vergeleken. Daar waren grote verschillen, het meest dramatisch

was een zelfbouw programma dat een product-momentcorrelatie gaf in de buurt van 3. En dat terwijl die toch echt tussen -1 en 1 moet liggen. De oorzaak was dat programmeurs vaak niet meer deden dan formules uit een statistiekboek omzetten in Fortran-code zonder met dit soort zaken rekening te houden.

Gelukkig kwam er een pakket op de markt dat een grote hoeveelheid subroutines voor allerlei wiskundige en statistische functies bevatte die allemaal wél behoorlijk waren geprogrammeerd. Zoiets heette een subroutine-bibliotheek. Als dat pakket was geïnstalleerd kon je die subroutines als kant-en-klare bouwstenen direct 'linken' aan je eigen programma. Naast het verkrijgen van meer betrouwbare resultaten bespaarde je ook een hoop tijd en moeite.

Op een niet nader te noemen universitair rekencentrum was dit pakket aangeschaft en men had met veel publiciteit een presentatie georganiseerd. Daar kwam een groot aantal programmeurs van allerlei pluimage opdagen. De meesten kenden elkaar wel van contacten tijdens het aanleveren van de ponskaarten en het enkele uren later weer ophalen van de grote vellen kettingpapier met de uitkomsten. Maar er waren ook twee onbekenden aanwezig die wat bij de rest afstaken: duidelijk wat ouder en in een keurig pak gestoken. Dat bleken vertegenwoordigers van de Universiteitsbibliotheek te zijn. Zij begonnen vragen te stellen over de maximale uitleentijd, controle op het onbeschadigd terugbrengen, de officiële omschrijving van de routines volgens de standaardregels voor bibliotheken etc. Zij waren kennelijk naar de bijeenkomst gestuurd om die vrijgevochten cowboys van programmeurs eens goed uit te leggen hoe het écht in een bibliotheek behoort toe te gaan. Het heeft enige moeite gekost de

heren te overtuigen dat de benaming subroutine-bibliotheek niets te maken had met het gebruikelijke werk van een klassieke bibliotheek...

Tja, dat waren nog eens pioniersdagen, mijmert deze opa...

GERRIT STEMERDINK is eindredacteur STATOR. E-mail: gjstemerding@hotmail.com



Illustratie: Cornelis Jetses