

Ruwe schats vertaalproces.

We zullen proberen om de vertaler in twee left to right scans in elkaar te draaien.ⁿ
In de eerste scan gebeurt onder meer het volgende.

De heptaden van de invoerstroam worden tot basic characters geassembleerd en in deze vorm wordt de text van het ALGOL programma ten bate van de tweede scan opgebouwd. In deze representatie komen enige non-ALGOL characters voor, zoals "Nieuwe Regel" en "illigitiem", bv. een rare onderlijning. We zullen nl. proberen om bij check alarm de pass in kwestie af te maken; het is niet gezegd, dat dat kan; immers, als je de executie van een programma beschouwt als "de laatste pass", wat het toch eigenlijk is,, dan hebben we inmiddels al besloten, dat we in geval van alarm zullen stoppen.

Behalve, dat de text in basic characters voor later gebruik wordt opgebouwd, wordt hij in de eerste scan aan een aantal analyses onderworpen. Ten eerste zullen we de delimiter structure zorgvuldig checken; in deze phase zullen we nooit gebruik maken van "de bijbehorende declaratie" omdat die in principe nog ongezien kan wezen.

Verder is de belangrijkste taak van de eerste scan om de volledige naamlijst op te bouwen. Wij stellen ons voor, dat dit gaat via een stapel, die als transition mechanism gebruikt wordt: hierin worden alle gedeclareerde namen plus de uit deze declaratie volgende en voor de tweede pass relevante gegevens opgenomen. Zodra een blok einde gedetecteerd is, wordt de top van deze naamstapel naar de definitieve naamlijst overgeheveld. Wel moesten we door enige ketening zorgen, dat de boomstructuur van deze naamlijst correct tot uiting gebracht wordt. In deze definitieve naamlijst fokken we dus de namen blokgewijze in volgorde van blokeinde. Hiermee hebben we bereikt, dat voor dit naamsorteerproces elke naam precies een keer verhuist wordt. Dit is mooi.

Nevenzaak van de eerste scan is om de niet lege regels van de aangeboden text te nummeren en om via de output op herkenbare plaatsen het toegekende regelnummer uit te printen. (Bv. in regels, waarin labels voorkomen en regels, waarin procedures gedeclareerd worden. De output mag zich hier bedienen van een enkel alfabet.) In deze output mag zelfs wel iets van de blokstructuur tot uiting komen; als in de eerste scan al fouten ontdekt worden, komen deze foutmeldingen door dit lijstje heen, maar dit achten we geen bezwaar (we = Cor en ik).

In de tweede scan wordt het complete ALGOL-programma definitief gemaakt en op de segmenten ingedeeld. Dit lijkt een beetje moeilijk, maar moet wel meevallen, want we hebben drie machtige kanonnen tot onze beschikking. Ten eerste hebben we de volledige naamlijst met alle declaratiegegevens, die we maar wensen (daar had de eerste pass dan maar voor moeten zorgen). Ten tweede is dit essentieel een vertaler, die het object-programme in het geheugen opbouwt, ten derde hebben we de volledige text van het object-programme ook al in het geheugen staan en kunnen we de vertaler tamelijk straffeloos een stukje in de toekomst laten kijken.

Een en ander impliceert wel, dat de routine "read next character" een soortement van juweeltje dient te worden. Ten eerste kent hij twee toestanden: in de eerste fase moet hij het volgende basic character uit de heptaden opbouwen en dumpen, in de tweede fase moet hij "the next character" uit de gedumpte informatiestroom halen. Ter verhoging van de feestvreugde zal hij wel twee pointers moeten hebben, een testpointer en een gehadpointer.

Dit lijkt in deze staat van "ruwheid" een redelijk overzicht van de taken.

8 december 1964