

M. ZRIMEC, B. VILFAN, R. DORN  
FAKULTETA ZA ELEKTROTEHNIKO  
LJUBLJANA

SAOPŠTENJE

PRISTOP K IMPLEMENTACIJI JEZIKA PASCAL NA RAČUNALNIK IBM 1130

AN APPROACH TO IMPLEMENTATION OF PASCAL ON THE IBM 1130  
COMPUTER

V referatu je opisan pristop k implementaciji programskega jezika Pascal P na računalnik IBM 1130. Opisani so problemi, s katerimi smo se srečali in predlagane rešitve.

The paper describes the first approach to an implementation of Pascal P on the IBM 1130 computer. The chosen machine places severe constraint on the computer writer, mainly with regard to the memory size. Thus, there are numerous problems to be solved during an implementation, and these problems, as well as the proposed solutions, are discussed.

#### UVOD

Še pred desetimi leti so pisanje računalniških programov primerjali z umetnostjo, danes pa postaja to vedno bolj inženirsko delo. Prav gotovo so bistven delež k tej spremembi mišljenja prispevali moderni programski jeziki, ki nam poleg vseh drugih prednosti, ki jih dajejo tudi lažji in bolj sistematičen pristop k učenju vsake zahtevne vede, kot je programiranje.

Še pred nekaj leti se je na naši fakulteti slišalo, da se računalništvo in inženiringe obravnavata v program-

skem jeziku Pascal /3/. Rezultati učenja so odlični, če jih primerjamo s slušatelji, ki se učijo ostale programske jezike na primer Fortran. Do istih rezultatov so prišli predavatelji programiranja na srednjih šolah. Najboljše potrdilo te trditve pa so rezultati prvega republiškega tekmovanja srednješolcev v računalništvu in informatiki, ki ga je organizirala naša fakulteta lansko leto: prva mesta so, razen izjem, zasedli dijaki, ki so poznali programski jezik Pascal.

Na žalost pa prevajalnik za Pascal ni implementiran na računalniku IBM 1130, ki ga imamo v računskem centru naše in še marsikatero fakultete v Jugoslaviji.

Nadaljevanje tega referata je prvi pristop k tej implementaciji.

#### PASCAL P /1/

Programski jezik Pascal, oziroma njegov prevajalnik, služita za osnovo prenosljivega prevajalnika z imenom Pascal P. Pascal P prevaja podmnožico jezika 'Standard Pascal' in je ves napisan v jeziku, ki ga prevaja. Tudi sam jezik, ki ga prevajalnik prevaja, imenujemo Pascal P. Razlike med 'Standard Pascal-om' in Pascal P-jem so minimalne:

- a.) Pascal P ne pozna procedur in funkcij kot parametrov,
- b.) Skočni stavki (goto) ne smejo voditi izven bloka procedure ali funkcije,
- c.) Pascal P pozna samo predefinirane znakovne datoteke (input, output),
- d.) Ključna beseda "packing" je brez pomena in
- e.) V Pascal P-ju ne poznamo standardne procedure "dispose" pač pa dve novi proceduri 'mark' in 'release'.

Vse ostale karakteristike in lastnosti Pascal P-ja so enake standardnemu Pascal-u, zato jih ne bi posebno naštevati.

KRATEK OPIS KOMPILATORJA IN INTERPRETORJA /1/

Od avtorjev Pascal-a P-ja /1/ smo prejeli paket za implementacijo tega jezika, ki je vseboval kompilator in interpretor. Oba programa sta napisana v standardnem Pascal-u. Za tako verzijo implementacijskega paketa smo se odločili zato, ker imamo dostop do pravejalnika za Pascal na Republiškem računskem centru.

Kompilator Pascal P-ja prevede vhodni program v zbirni jezik za namišljeni skladovni računalnik sc. Kompilator uporablja za prevajanja metodo rekurzivnega sestopa in se zelo malo razlikuje od Pascalovega kompilatorja. Praktično so avtorji spremenili samo zadnja dva razvojna koraka kompilatorja za Pascal, to je generacijo kode.

Interpreter služi predvsem opisu skladovnega računalnika sc. Ima dva dela. Prvi je enostavni zbirnik, ki prevede mnemonično kodo v strojni jezik sc-ja. (Izhod kompilatorja je namreč znakovna datoteka.) Drugi del pa tako dobljene ukaze interpretira.

SKLADOVNI RAČUNALNIK sc /1/

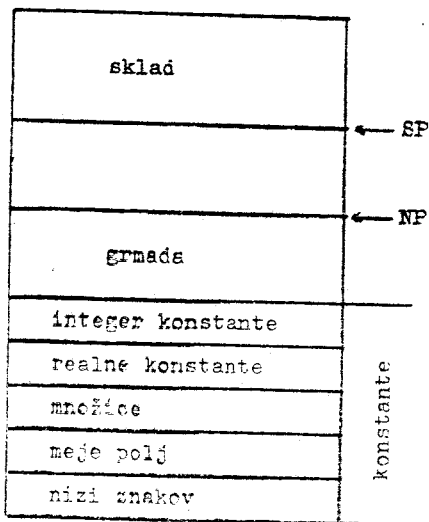
Skladovni računalnik sc ima pomnilnik in 5 registrov:

- PC programski števec
- SP kazalec na vrh sklada
- MP kazalec za 'mark'
- NP kazalec za 'new'
- EP kazalec 'zasedenosti' sklada

Pomnilnik delimo logično v dva dela:

- pomnilnik, ki vsebuje ukaze programa - CODE
- pomnilnik, ki vsebuje podatke - STORE

CODE ima tri polja: OP, P in Q.  
 STORE je še nadalje razdeljen v dva dela: prvi del vsebuje različne konstante, drugi del pa zadošča dinamičnim potrebam za pomnilniški prostor, kot to zahteva izvajanje Pascal-ovih programov (glej sliko).



### RAČUNALNIK IBM 1130 /4/ /5/

Računalnik IBM 1130 na naši fakulteti ima 16K 16 bitnih besed in 6 osnovnih registrov:

- A akumulator
- 4 podaljšek akumulatorja
- I instrukijski register
- XR1
- XR2 trije indeksni registeri
- XR3

Konfiguracija celotnega sistema je sledeča:

- centralno enoto s pomnilnikom
- operatorjevo konsolo
- dve enoti za disk
- čitalnik in luknjalnik kartic
- hitri tiskalnik in
- risalnik

Zelo velika pomanjkljivost tega računalnika je togost operacijskega sistema (monitorja) in nezadostna dokumentacija. Operacijski sistem vsebuje tri prevajalnike. To so zbirnik oziroma makrozbirnik, RPG in Fortranski kompilator. Vsi prevajalniki lahko sprejemajo vhodni program samo na podatkovnih karticah. Prevajalnikov tudi ne moremo spremeniti tako, da bi vhodni program čitali z diska, ker nimamo ne odgovarjajočih listingov niti opisov programov.

#### KRIŽNI PREVAJALNIK

Najenostavnejši način za implementacijo Pascala P na IBM 1130 je križni prevajalnik. Na izbiro imamo dve varianti:

- 1.) Kodo sc-ja interpretiramo ali pa
- 2.) Kodo sc-ja prevedemo v strojne ukaze za IBM 1130.

Odločili smo se za drugo varianto, ki je vmesna točka za dosego našega cilja. Za prevajanje uporabljamo makro zbirnik, zato je potrebno delno spremeniti prevajalnik za Pascal P. Predvsem je potrebno popraviti obliko mnemoničnih ukazov sc-ja, ker so ukazi makro zbirniku na IBM 1130 formatizirani. Za vsak ukaz sc-ja naj bomo semantično enakovreden makro in le tega shranimo v makro knjižnici.

Primer: ukaz NOT ("boolean not"):

- indekсни register 1 kaže na vrh sklada
- v TABLE-1 imamo konstante 1
- Vrednost "TRUE" predstavimo z 1

IVI.368

Vrednost "FALSE" predstavimo z 0

```
MACRO
NOT
LD TABLE+1
EOR L1 0
STC L1 0
MEND
```

Opomba: v naboru ukazov IBM 1130 ni ukaza NOT.

Pri implementaciji križnega prevajalnika naletimo na nekaj manjših problemov pogojenih predvsem s premalo dolgo strojno besedo (16 bitov).

Za realna števila in množice ta dolžina ne zadošča. Zato smo vzeli za realna števila dve zaporedni besedi (standardna natančnost), za množice pa tri zaporedne besede. Predpostavljamo, da to zadošča za potrebe študentskih programov.

Križni prevajalnik je seveda lahko samo vmesna stopnja pri implementaciji Pascala P na IBM 1130, ker ima več slabih kot dobrih lastnosti:

- prvič je makro zbirnik izredno počasen in kot tak popolnoma neuporaben za študentske programe, kjer je bistven čas prevajanja in manj izvajanja programa,
- drugič prevajalnik iz Pascal P-ja v kodo za sc je napisan v Pascalu in se torej izvaja na računalniku, ki že ima prevajalnik za Pascal in
- tretjič prevajalnik iz Pascal P-ja v kodo za sc, napisan v kodi za sc, bi zavzel več pomnilniškega prostora kot ga imamo.

Po drugi strani pa lahko s križnim prevajalnikom pripravi-

mo in preverimo generacijo kode za končno verzijo prevajalnika. Vsak makro lahko namreč dobesedno prepisemo v generacijske rutine samega prevajalnika in se na ta način izognemo makro zbirniku.

Še vedno pa ostaja problem pomanjkanja pomnilniškega prostora.

### POGLAVJA PREVAJALNIKA

Prevajalnik za Pascal P je potrebno razdeliti na več delov, ki medsedbojno komunicirajo s pomočjo skupnega dela pomnilnika in vmesne kode vsakega poglavja prevajalnika.

Skupni del pomnilnika vsebuje tabela, ki nastajajo in so potrebne pri prevajanju vhodnega programa (na primer: simbolna tabela).

Vsako poglavje prevajalnika se vloži in izvaja posebej. Vmesne kode so na diskih, zato je primerno, da uporabimo oba diska računalnika IBM 1130. Zaradi hitrejšega prevajanja je umestno, da je disk, na katerem je vhodna vmesna koda za neko poglavje, različen od diska, ki nosi izhodno vmesno kodo tega poglavja. Pri naslednjem poglavju pa se ti vhodno izhodni funkciji obeh diskov zamenjata.

Prevajalnik razdelimo na pet delov:

- leksikografska analiza, ki pregleda vhodni program znak za znakom in prevede ključne besede, imena in konstante v cela števila.
- sintaktična analiza preveri, če je program sintaktično pravilen.
- analiza deklaracij preveri vse deklaracije in v simbolno tabelo zapiše attribute simbolov.

### III.370

- analiza stavkov preveri kompatibilnost operandov in operatorjev.
- generacija kode generira strojne ukaze računalnika IBM 1130.

Po približnih ocenah lahko predvidevamo, da bodo različna poglavja prevajalnika zasedla največ do 12K pomnilnika. (glej tabelo)

- leksikografska analiza	4 K
- sintaktična analiza	12 K
- analiza deklaracij	10 K
- analiza stavkov	2 K
- generacija kode	4 K
- skupni pomnilnik	9 K

Tabela približnih ocen zasedbe pomnilnika

#### ZAKLJUČEK

Programski jezik Pascal, vsaj kolikor nam je znano, še nikjer na svetu ni bil implementiran na računalniku IBM (podatek iz leta 1977), čeprav bi bil za študentske potrebe zelo koristen. Implementacija Pascal P-ja pa je prvi korak v tej smeri.

#### LITERATURA

- /1/ K.H.Novak, U.Aumann, L.Jensen, R.H.Nardi, Ch.Lacoti:  
The Pascal P Compiler: Implementation Notes,  
July 1979, Muenchen, Technische Universität,  
Deutsch.



- /2/ N.W.RTH Algorithms + data structures = programs.  
Prentice-Hall, New Jersey, 1976.
- /3/ Pascal User Manual and Report. K.Jensen, N.Wirth,  
Springer-Verlag, N.Y. 1975.
- /4/ IBM Systems Reference Library. IBM 1130/1800 Assemb-  
ler Language.
- /5/ IBM 1130 Subroutine Library.

