

Ladislav Koubek

Vytváření seznamů v translátoru z jazyka Algol 60

Acta Universitatis Carolinae. Mathematica et Physica, Vol. 10 (1969), No. 1-2, 103--104

Persistent URL: <http://dml.cz/dmlcz/142239>

Terms of use:

© Univerzita Karlova v Praze, 1969

Institute of Mathematics of the Academy of Sciences of the Czech Republic provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This paper has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* <http://project.dml.cz>

VYTVÁŘENÍ SEZNAMŮ V TRANSLÁTORU Z JAZYKA ALGOL 60

L. KOUBEK

Centrum numerické matematiky UK, Praha

ОБРАЗОВАНИЕ МАГАЗИНОВ В ТРАНСЛЯТОРЕ С ЯЗЫКА АЛГОЛ 60. В работе приводится система работы с магазинами при компиляции в трансляторе с языка АЛГОЛ 60 и указывается система автоматического распределения памяти, осуществляемая в течение работы машинного кода, созданного компилятором.

Bloková struktura textu psaného v ALGOLu 60 ovlivňuje jak vlastní překladač, tak program jím vytvořený.

Při práci překladače musíme seznamy podle popisů vytvářet v závislosti na blokové struktuře textu, neboť identifikátor může být použit v různých blocích v různém významu.

Každému bloku přiřadíme zřejmým způsobem stupeň a šest pracovních polí, do kterých při kompilaci zapisujeme stav indexů jednotlivých seznamů v okamžiku vstupu do bloku. Podle popisů pak tyto seznamy doplňujeme. Po nalezení symbolu end, který je ukončujícím symbolem bloku (ve smyslu jednotky programu), jednoduše obnovíme stav všech indexů seznamů, čímž automaticky vymažeme údaje zařazené do seznamů v tomto bloku.

Vyhledávání v seznamech musíme ovšem organizovat tak, že prohlédneme nejprve všechny seznamy vytvořené v bloku nejvyšší úrovně; nenajdeme-li v nich identifikátor, prohlédneme seznamy bloku nižší úrovně atd. Teprve když nenajdeme identifikátor ani v seznamech bloku stupně 0, zařadíme identifikátor do seznamu návěští.

Při tomto postupu nemusí dojít k chybě, uvedeme-li některý identifikátor v několika popisech téhož bloku. Kompilátor nalezne vždy ten identifikátor, který byl uveden později, nebo který je zařazen v seznamu prohlíženém dříve.

Jedinou potíží je práce s návěštími. Návěští nejsou popisována a identifikátory musíme do seznamu návěští zařazovat tehdy, kdy se poprvé vyskytnou v textu. Adresu skoku jim ovšem můžeme přiřadit teprve, když je tímto návěštím označen některý příkaz.

Je-li identifikátor návěští cílovým výrazem příkazu skoku na příkaz, který je zapsán později, nevíme (v okamžiku kompilace), zda půjde o skok do bloku nebo do nadbloku. Musíme proto identifikátor zařadit do seznamu návěští i tehdy, když je už uveden v seznamu v nadbloku.

Při nalezení ukončujícího symbolu bloku musíme projít celý seznam návěští. Návěští, kterým jsou už v bloku nejvyššího stupně přiřazeny adresy, musíme vymazat a ostatním přiřadit adresy z bloku stupně o 1 nižšího.

Při práci strojového programu využíváme blokovou strukturu při umístování polí do paměti počítače. Na počátku programu každého bloku zařadíme instrukci, která hodnotu indexu bloku dosadí do zvláštního registru. Kromě toho v poli standardních pracovních buněk vyhradíme zvláštní pole, do kterého zapisujeme stav rozdělení paměti na začátku práce bloku. (Obvykle stačí zapsat jen poslední adresu obsazenou poli nadbloku). Při výpočtu linearizačních konstant a dimenzí polí zapisujeme nový údaj o rozdělení paměti do odpovídající buňky pole. Je zřejmo, že na konci bloku nemusíme zařazovat zvláštní program.

Pole popsaná v operační části procedury se vytvářejí ve fiktivních blocích přiřazených jednotlivým úrovním vyvolání procedury. Proto v každém pracovním poli procedury jednu buňku vyhražujeme pro údaj o rozdělení paměti. Tento údaj používáme stejným způsobem jako údaj o rozdělení paměti prováděním mimo proceduru.

Literatura

/1/ J. W. Backus - Programování v jazyku ALGOL 60, SNTL Praha
1963