Arithmetic Expression Compiler

Autor:
Teo Samaržija
(student FERIT-a)

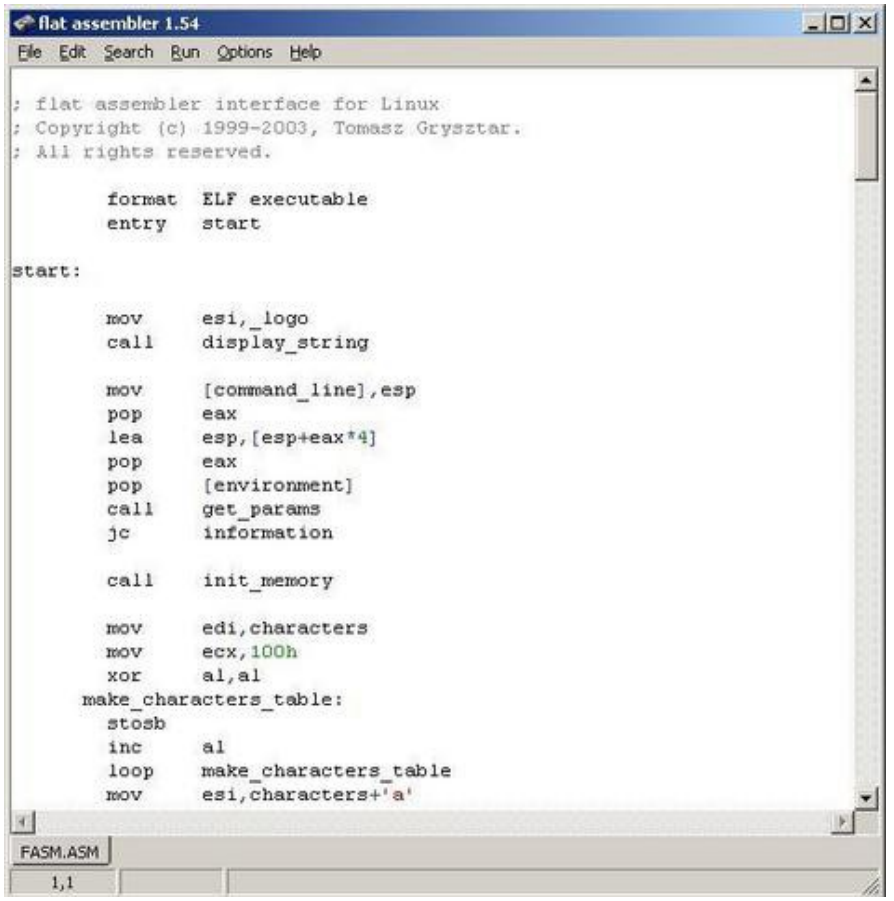
Arithmetic Expression Compiler dostupan je na ovoj web-adresi:

http://flatassembler.000webhostapp.com/compiler.html
(ili upišite u tražilicu "*Converting Arithmetic Expressions to Assembly*")

On pretvara aritmetičke izraze u assemblerski kod kompatibilan sa i486 procesorom.

Korišteni programi

- Notepad++ - uređivač programskog koda *(dobar za starija računala)*
- Internet Explorer 6 – testiranje JavaScript koda *(ako radi tamo, radit će svugdje)*
- Opera 36 – objavljivanje na internet *(za Windows XP)*
- Flat Assembler – testiranje assemblerskog koda *(moćan pretprocesor, jednostavni sintaksa i IDE)*



Čemu služe compileri?

Važno za razumjeti:

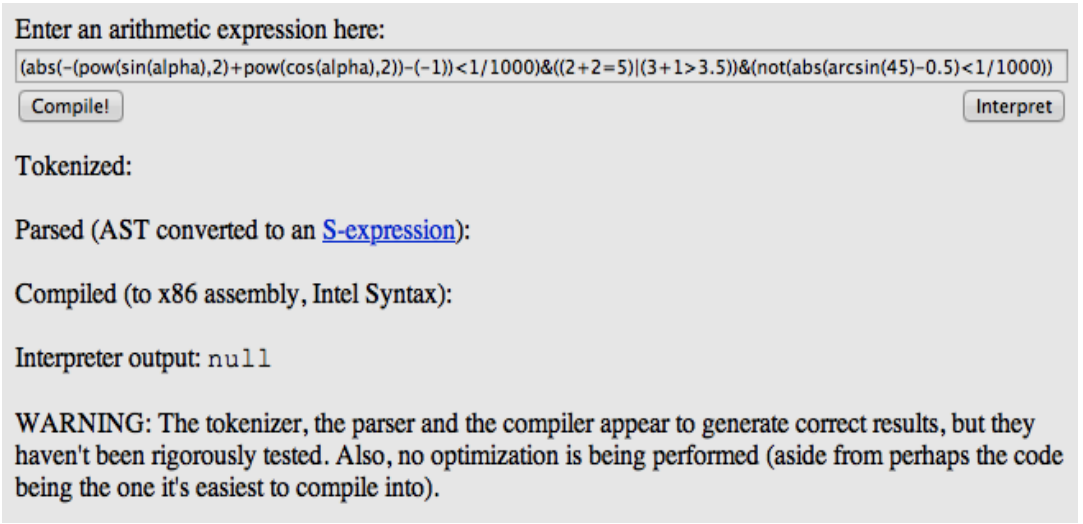
Računala ne govore različitim jezicima. Ona razumiju samo strojni jezik (nule i jedinice). Da bi razumjela programske jezike, potrebni su im posebni programi. Ti se programi zovu compileri i interpreteri.

Od čega se sastoji compiler?

1) Driver
2) Pretprocesor (nije nužan za sve jezike)
3) Thin-layersi (nisu nužni za sve jezike)
4) Tokenizer
5) Leksički analizer (obično dio tokenizera)
6) Parser
7) Syntax highlighter (ne mora komunicirati s ostalim dijelovima, no danas to obično čini)
8) Compiler u užem smislu (ili, danas rijetko, prevođenje programskog koda u strojni ovdje završava interpreter)
9) Assembler
10) Linker

Čemu služi driver?

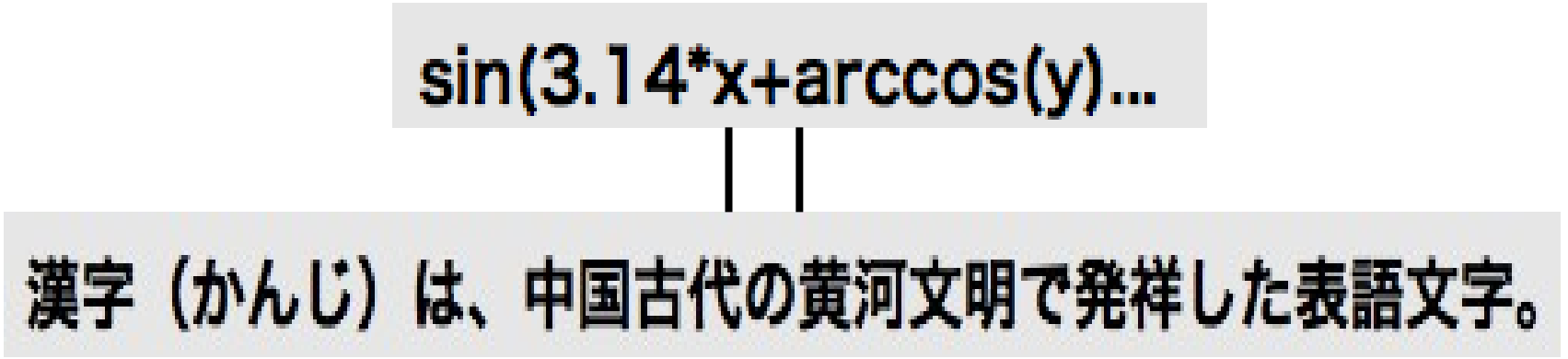
- Obično se prvi otvara i zadnji zatvara.
- Pokreće druge dijelove compilera i omogućuje komunikaciju među njima.
- Pruža sučelje (obično CLI) korisniku ili IDE-u.



Driver od *Arithmetic Expression Compilera*

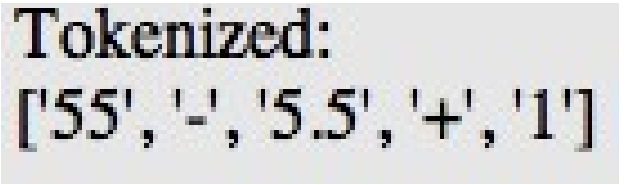
Čemu služi tokenizer?

- Netokenizirani ili neispravno tokenizirani tekst na programskom jeziku računalu izgleda ovako:



Gdje završava koja riječ?

- Programski jezici, kao i japansko pismo, imaju mehanizme za razdvajanje riječi, ali oni nisu trivijalni.



Tokeniziran matematički izraz *55-5.5+1*

Čemu služi leksički analizer?

- On određuje vrste riječi, jer bez toga sintaktička se analiza ne može izvršiti.

Je li ovdje *flies* imenica ili glagol? Je li *like* veznik ili glagol?

- Programski jezici su obično takvi da se iz samog oblika riječi može odrediti koja je vrsta, bez poznavanja značenja ijedne riječi u rečenici i bez poznavanja sintaktičkih struktura.
- Važno je da parser može, na primjer, odmah reći je li nešto decimalni broj, da ne mora svaki put tražiti ima li decimalnu točku (to je posebno važno, jer je, primjerice, u C-u *2F* isto decimalni broj).

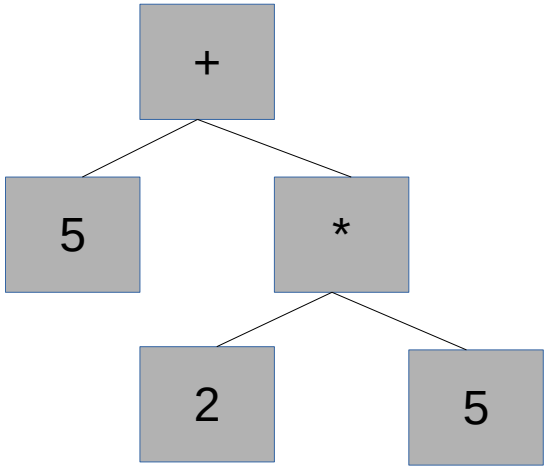
Čemu služi parser?

- Neparsirani ili neispravno parsirani tekst na programskom jeziku računalu izgleda ovako:

Ea, quae fertilissima totius Germaniae sunt, loca Graecis aliquibus nota fama esse loquuntur.

Je li *nota* vezano za *loca* ili za *fama*?

- Takve su situacije u programskim jezicima veoma česte, jer u programskim jezicima, osim u iznimnim slučajevima, ne postoji ekvivalent deklinacijama i konjugacijama.
- Parser to rješava gradnjom AST-a (Abstract Syntax Tree).



AST od *5+2*5*

Parsed (AST converted to an S-expression):
(+ 5 (* 2 5))

S-expression – jedan od načina da se AST prikaže kao niz znakova

Čemu služi *syntax highlighter*?

- On boja različite vrste riječi u programskim jezicima različitim bojama. Smatra se da je tako tekst na programskom jeziku lakši za čitanje.
- BONUS: Microsoft Edge ima tu mogućnost za engleski jezik, jer to navodno ubrzava čitanje onima koji slabije znaju engleski.

```
finit
mov [result],2f
fld dword [result]
mov [result],2f
fld dword [result]
faddp st1,st0
mov [result],5f
fld dword [result]
fcomip st1
fstp dword [result]
jna l26167
fld1
jmp l2661
l26167:
fldz
l2661:
fstp dword [result]
```

Sintaksno obojeni assemblerski tekst. Plavo su mnemonike (glagoli), crveno varijable, zeleno brojevi, ružičasto pridjevi, tirkizno registri, a žuto oznake za preskakanje (*labels*).

Compiler u užem smislu i interpreter

- Compiler zamjenjuje riječi iz AST-a njihovim prijevodima na strojni jezik, pri tome za gotovo sve naredbe koristi kratice (iz assemblerskog jezika) kako bi olakšao traženje grešaka. Program se kasnije može assemblirati i pokrenuti.
- Interpreter riječi iz AST-a odmah prevede na strojni jezik (ili drugi interpretirani jezik, kao što je JavaScript) i izvede ih (što zahtijeva da i računalo na kojem će se program izvršavati ima interpreter i obično se vrti sporije od compiliranog programa).

Kako izgleda programski kod jednostavnog compilera?

```
}
function fasinp() //-||- arkus sinus (po formuli arctan(sin(x)/sqrt(1-sin(x)*sin(x))))
{
    asm("fstp dword [result]");
    asm("fld dword [result]");
    asm("fld1");
    asm("fld dword [result]")
    asm("fld dword [result]");
    fmulp();
    fsubp();
    fsqrt();
    fdivp();
    fatanp();
}
function facosp() // -||- arkus kosinus (po formuli pi/2-arcsin(x))
{
```

Čemu služi assembler?

- Neassemblirani ili neispravno assemblirani tekst na programskom jeziku računalu izgleda otprilike kao što tekst na internetskom forumu izgleda čovjeku neupučenome u taj vokabular.

BTW, IMHO, U R taking it 2 serious.

Assemblerski jezik je, sa svim svojim kraticama, prema strojnom jeziku isto što je i ovo prema engleskom jeziku.

- Assembler je program koji kratice iz assemblerskog jezika zamijenjuje nulama i jedinicama.

Čemu služi linker?

- Linker je rječnik manje poznatih riječi iz programskog jezika, kako compiler u užem smislu ne bi morao sve znati.
- U mnogim programskim jezicima moguće je tvoriti nove riječi te ih spremati u *librariese* i koristiti u više programa. Linker je program koji zna čitati te *librariese*.

```
finit
mov [result],5f
fld dword [result]
mov [result],5f
fld dword [result]
faddp st1,st0
;Here the linker should insert the code for calling the function 'fib'!
fstp dword [result]
```

Ovo ispisuje compiler za *fib(5+5)*, jer *fib* nije definiran.

Zadaci

1) Provjeri pomoću AEC-a koji je od sljedećih izraza na njegovom jeziku ispravan:

   a) 5--

   b) 5**5

   c) (5*-2)(-5*2)

2) Koji dio compilera javlja grešku ako mu se ukuca:

   a) 5+3.1.4

   b) 2b+not(2b)

   c) 1+2)+3+4*(

   d) f(5+5)

3) Skiciraj AST od *(5+2)*5*

4) Poredaj compilirani kod od izraza po veličini:
   a) sqrt(123)
   b) 5+5
   c) arcsin(0.5)
   Što misliš, zašto je rezultat takav?

5) Ako ukucate izraz poput *2&3,* compiler će ispisati poruku o pogrešci, ali će svejedno ispisati i neki assemblerski kod. Pokušaj objasniti zašto.

Korištena literatura

- Assembly Language for x86 Processors, Kip Irvine, 1999. *(Assembler)*
- w3schools.com *(JavaScript)*

Hvala na pozornosti!