

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

1. Úvod	2
1. Skúška	3
A skupina.....	3
Prvý príklad	3
Druhý príklad	3
B skupina.....	3
Prvý príklad	3
Druhý príklad	4
2. Skúška	4
A skupina.....	4
Prvý príklad	4
Druhý príklad	5
B skupina.....	6
Prvý príklad	6
Druhý príklad	7
3. Skúška	8
A skupina.....	8
Prvý príklad	8
Druhý príklad	8
B skupina.....	8
Prvý príklad	8
Druhý príklad	10

V tomto dokumente sú príklady na skúškach z predmetu Prekladače v školskom roku
2002/2003 - letný semester

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

Úvod

Keďže príklady z jednotlivých skúšok sú rovnaké, tak som z istých dôvodov riešil iba príklady z druhej skúšky. Sčasti je riešený aj jeden príklad z tretej skúšky, kde trebalo nakresliť derivačný strom.

doc. Kollár sa na skúške zameriaval hlavne na gramatiku v prvom príklade a na syntaktický analyzátor so zotavením v druhom. Na teóriu sa veľmi nezameral, bolo iba zopár vyvolených ľudí, ktorým dával aj teóriu. S toho čo som počul tak sa pýtal na: **derivačný strom, aktivačný záznam, definícia LL(1)** a iné o ktorých zatiaľ neviem.

Mňa osobne spradol zato, že sa neviem odborne vyjadrovať a že ak sa takto budem vyjadrovať na štátnici tak.....potom ma vyhodil ;(

Robiť ťaháky nemá veľký zmysel, pretože doc. Kollár vymýšľá príklady na začiatku každej skúšky. **Stále sú iné!**

Za prípadne chyby ktoré som narobil sa ospravedlňujem a prosím čitateľa aby mi ich oznámil na frend@szm.sk

1. Skúška

A skupina

Prvý príklad

Navrhnuť gramatiku zátvorkovaných výrazov a prekladač do postfixného tvaru pre:

N +

L -

P *

$p(+) > p(-) > p(*)$, id: cdbcedd

pozn. "id: cdbcedd" je tvar identifikátora, teda naznačuje nám to že identifikátor je zložený z písmen. "id" môže mať ľubovoľný tvar, napr. id: iadufai, id: a, id: abc, ...

"N +" znamená že operácia + je neasociatívna

"L -" znamená že operácia - je asociatívna zľava

"P *" znamená že operácia * je asociatívna sprava

napr. majme výraz $a+b+c$. Pri pravej asociatívnosti sa vyhodnotí ako $(a+(b+c))$, pri ľavej $((a+b)+c)$

$p(+)$, $p(-)$, $p(*)$ udávajú prioritu operácii

Na prepis tejto neformálnej špecifikácie na gramatiku existujú určité **pravidlá**:

Pravidlo	Asociatívnosť
$A \rightarrow B [\oplus B]$	žiadna
$A \rightarrow B [\oplus A]$	pravá
$A \rightarrow B \{ \oplus B \}$	ľavá

Postfixný tvar : operátor je za operandami

napr.

+ 5 5	prefixný tvar
5 + 5	infixný tvar
5 5 +	postfixný tvar

Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:

$A \rightarrow a[B] (c | D)$

B skupina

Prvý príklad

Navrhnuť gramatiku zátvorkovaných výrazov a interpretátor

P+

L *

N /

$p(/) > p(*) > p(+)$, const: 125..

Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:

$A \rightarrow a[B] (c | D)$

2. Skúška

A skupina

Prvý príklad

Navrhnuť gramatiku zátvorkovaných výrazov a prekladač do postfixného tvaru pre:

$L >$

$P <$

$N =$

$p(>) > p(<) > p(=)$, id: cdbcd

Riešenie:

$A \rightarrow B ["=" B]$
 $B \rightarrow C ["<" B]$
 $C \rightarrow D \{ ">" D \}$
 $D \rightarrow \text{string} | "(" A ")"$
 $\text{string} \rightarrow (a|...|z)\{a|...|z\}$

PROCEDURE A

```
BEGIN
  B;
  if symbol=eq then
    BEGIN
      getsymbol;
      B;
      print("=");
    END
  END
END
```

PROCEDURE C

```
BEGIN
  D;
  while symbol=great then
    BEGIN
      getsymbol;
      D;
      print(">");
    END
  END
END
```

PROCEDURE B

```
BEGIN
  C
  if symbol = small then
    BEGIN
      getsymbol;
      B;
      print("<");
    END
  END
END
```

PROCEDURE D

```
BEGIN
  case symbol
    in string:
      BEGIN
        print(TID[ival]);
        getsymbol;
      END
    in lpar:
      BEGIN
        getsymbol;
        A;
        getsymbol;
```

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

END

END

Translačná schéma:

Lex[=] = eq
Lex[>] = great
Lex[<] = small
Lex[(] = lpar
Lex[)] = rpar
Lex[c₁...c_n] = string<ival>
n ≥ 1
ci patri do {a,...,z}
pričom TID[ival]="c₁...c_n"

pozn. premenná **symbol** obsahuje kód symbola načítaného procedúrou **getsymbol**. Procedúra **getsymbol** predstavuje lexikálny analyzátor. **TID[]** predstavuje pole identifikátorov. Bunky tohto poľa sú buď prázdne, alebo sú naplnené reťazcom. Napĺňanie poľa **TID[]** zabezpečuje procedúra **getsymbol**.

Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:

$X \rightarrow (Sa[D] \mid c) B$

Riešenie:

PROCEDURE X(K:symbolset)

BEGIN

check(1,H(S) U H(c) U H(B) U K);

case symbol:

in H(S):

BEGIN

S(H(a) U H(D) U H(B) U K);

if symbol = 'a' then getsymbol;

else error(2, H(D) U H(B) U K);

if symbol IN H(D) then D(H(B) U K);

END

in H(C):

BEGIN

if symbol='c' then getsymbol;

else error(3,H(B) U K);

END

else error(4, H(S) U H(c) U H(B) U K); /* tento error patri ku **case** */

B(K);

END

PROCEDURE error(n:INTEGER, K:symbolset)

BEGIN

printerr(n); /* vypise chybu na obrazovku/do súbora */

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

```
while NOT symbol IN K then getsymbol;  
END
```

```
PROCEDURE check(n:INTEGER, K:symbolset)  
BEGIN  
  if symbol NOT IN K then error(n,K);  
END
```

pozn. množiny $H(S), H(D), H(B)$ nie sú známe, pre tento príklad nie sú podstatné. Pre množiny $H(c), H(a)$ platí $H(c)=\{c\}$, $H(a)=\{a\}$

Procedúra **printerr(n)** vypisuje na obrazovku chybovú hlášku. Typ hlášky závisí od čísla **n**. napr. "Chyba 3: Očakával sa identifikátor!"

B skupina

Prvý príklad

Navrhnuť gramatiku zátvorkovaných výrazov a interpretátor pre celé čísla

P +
N -
N /
 $p(-) = p(/) > p(+)$, const: 123..

Riešenie:

A → B ["+"A]
B → C [("-"| "/")C]
C → const | "(" A ")"
const → (0|...|9){ 0|...|9}

```
PROCEDURE A  
VAR h1,h2:INTEGER  
BEGIN  
  B;  
  if symbol = plus then  
    BEGIN  
      getsymbol;  
      A;  
      h2=pop;  
      h1=pop;  
      push(h1+h2);  
    END  
END
```

```
PROCEDURE B  
VAR h1,h2:INTEGER  
BEGIN
```

```
  C;  
  case symbol  
  in minus:  
    BEGIN  
      getsymbol;
```

```
  C;  
  h2=pop; h1=pop;  
  push(h1-h2);  
END  
  
in div:  
  BEGIN  
    getsymbol;  
    C;  
    h2=pop; h1=pop;  
    push(h1/h2);  
  END  
  
END
```

```
PROCEDURE C  
BEGIN  
  case symbol  
  in const:  
    BEGIN  
      push(TC[ival]);  
      getsymbol;  
    END  
  in lpar:
```

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

```
BEGIN                                     END
getsymbol;                             END
A
getsymbol;
```

Translačná schéma:

Lex[+] = plus
Lex[-] = minus
Lex[/] = div
Lex[(] = lpar
Lex[)] = rpar
Lex[c₁...c_n] = const<ival>
n ≥ 1
ci patri do {0,...,9}
pričom TC[ival] = c₁...c_n

pozn. premenná **symbol** obsahuje kód symbola načítaného procedúrou **getsymbol**. Procedúra **getsymbol** predstavuje lexikálny analyzátor. **TC[]** predstavuje pole konštánt. Bunky tohto poľa sú buď prázdne, alebo sú naplnené konštantou. Napĺňanie poľa **TC[]** zabezpečuje procedúra **getsymbol**.

Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:
 $Y \rightarrow \{ a B \} \mid b B$

Riešenie:

PROCEDURE Y(K:symbolset)

BEGIN

check(1, H(a) U H(b) U K);

case symbol

in H(a):

BEGIN

check(2, H(a) U H(B) U K)

while symbol IN [H(a) U H(B)]

BEGIN

if symbol='a' then getsymbol;

else error(3, H(a) U H(B) U K);

B(H(a) U H(B) U K); /* tento riadok je spravny !! */

check(2, H(a) U H(B) U K);

END

END

in H(b):

BEGIN

if symbol='b' then getsymbol;

else error(4, H(B) U K);

B(K);

END

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

```
else error(5, H(a) U H(b) U K); /* tento error patri ku case */  
END
```

PROCEDURE error(n:INTEGER, K:symbolset)

BEGIN

```
  printerr(n); /* vypise chybu na obrazovku/do súbora */
```

```
  while NOT symbol IN K then getsymbol;
```

END

PROCEDURE check(n:INTEGER, K:symbolset)

BEGIN

```
  if symbol NOT IN K then error(n,K);
```

END

pozn. množina $H(B)$ nie je známa, pre tento príklad je nepodstatná. Pre množiny $H(a)$, $H(b)$ platí $H(a)=\{a\}$, $H(b)=\{b\}$

3. Skúška

A skupina

Prvý príklad

Interpretátor jazyka zátvorkovaných výrazov, kt. obsahuje operandy a operácie:
Neformálna špecifikácia:

```
"abcd" ++ "ef" => "abcdef" /* zret'azenie*/  
"abcd" = "abef" => "ab"    /* spoločný podvýraz */
```

$L ++$

$N =$

$p(=) > p(++)$

pozn. Tento príklad, tak ako aj predošlé pozostáva s návrhu gramatiky zátvorkovaných výrazov a návrhu interpretátora.

Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:

$A \rightarrow aB \{ cD \} d$

B skupina

Prvý príklad

Navrhните prekladač do jazyka zátvorkovaných výrazov a nakreslite derivačný strom pre výraz $a+b+c=d=e$

$P +$

$L =$

$N *$

$p(+) > p(=) > p(*)$, id: abcd..

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

Riešenie:

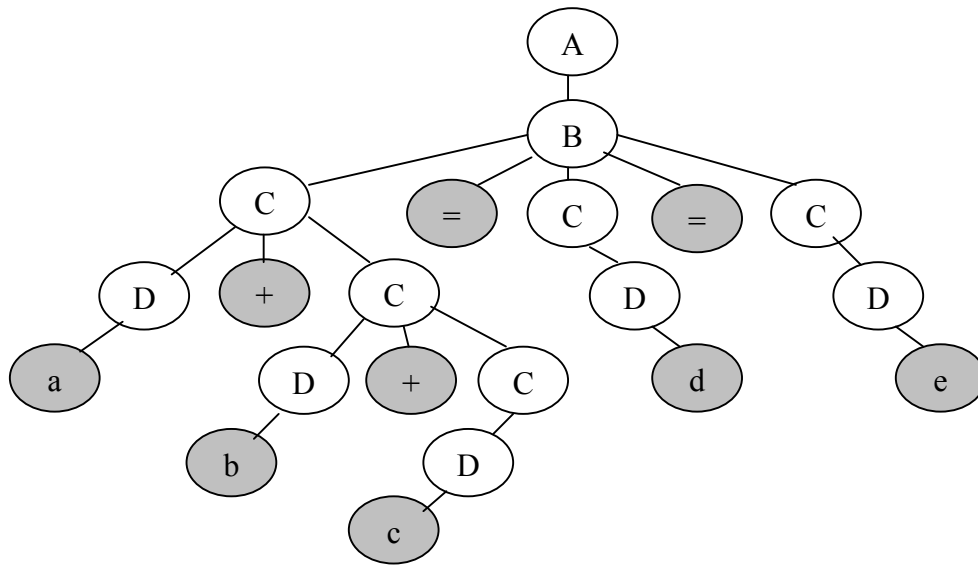
Keďže je tento príklad predsa trochu iný (derivačný strom), tak ho s časti vyriešim:

$$\begin{aligned} A &\rightarrow B \text{ ["*" B]} \\ B &\rightarrow C \text{ { "=" C } } \\ C &\rightarrow D \text{ ["+" C]} \\ D &\rightarrow \text{string} \mid \text{" (" A ")"} \\ \text{string} &\rightarrow (a|\dots|z) \{ a|\dots|z \} \end{aligned}$$

Štartovací symbol gramatiky je symbol A. Teda koreňom stromu bude symbol A!

Príklady na skúškach z predmetu Prekladače
18.6.2003 by Frend, frend@szm.sk

Derivačný strom pre výraz: $a + b + c = d = e$



Druhý príklad

Navrhnuť syntaktický analyzátor so zotavením pre:

$B \rightarrow (Ca \mid Bb) d$