

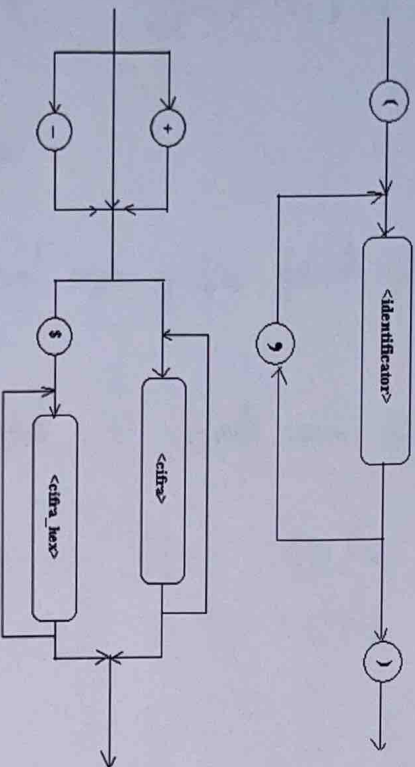
Seminar 1

Minilimbaje de programare. Identificarea și specificarea elementelor lexicale și sintactice.

1. Diagrame de sintaxă

Diagramele de sintaxă sunt reprezentări grafice folosite pentru a descrie sintaxa limbajelor de programare.

1.1. Dați câte 2 exemple valide care respectă următoarele specificații.



1.2. Descrieți, folosind diagrame de sintaxă, sintaxa secțiunii de declarații de variabile în Pascal. "Neterminalele" (vor apărea în dreptunghiuri cu colțurile rotunjite și) vor fi: <ID> și <type>.

2. BNF și EBNF

2.1. Dați o descriere echivalentă în BNF și EBNF corespunzătoare diagramei din secțiunea precedentă.

3. Elemente lexicale și sintactice ale unui limbaj de programare

3.1. Fie o gramatică ce descrie sintaxa unui mini-limbaj de programare, data prin regulile de producție:

```
<program>      → begin <lista_instr> end .
<lista_instr>   → <instr> ; <lista_instr>
<lista_instr>   → <instr>
<instr>         → <atribuire>
<instr>         → <instr_if>
<atribuire>     → ID = <expr>
<expr>         → <expr> + <variabila>
<expr>         → <variabila>
<variabila>    → ID
<instr_if>     → if ( <expr> ) then <atribuire>
```

- a) Identificați terminalele și neterminalele gramaticii
- b) Dați două "mini-programe" care sunt descrise de specificațiile date. ID este un atom lexical. Regulile de formare a acestui atom lexical ID (identificator), specificate folosind expresii regulate, sunt:
- $a(a|b|c)^*$

3.2. Fie următorul exemplu de program Pascal:

```
var f, a1, a2, a3 : integer;
begin
    a1 := 7;
    a2 := 11;
    a3 := a1 + a2 + 3;
    f := 5
end.
```

- a) Identificați elementele lexicale și structurile sintactice.
- b) Descrieți sintaxa structurilor sintactice folosind unul dintre mecanismele de specificare: BNF, EBNF sau gramatică independentă de context.
- c) Scrieți un program diferit de cel de mai sus care respectă descrierile date.
- d) Presupunând că operatorii și cuvintele cheie din exemplul de mai sus au asociate coduri – numere naturale în ordine crescătoare,
- descrieți conținutul tabelului FIP, precum și a tabelului de simboluri, atunci când se folosesc 2 tabele de simboluri, una pentru constante, una pentru identificatori, pentru fiecare dintre următoarele 3 organizări:
- Tabel sortat lexicographic
 - Arbore binar de căutare

iii.

Tabela de dispersie
(Alegeti o functie de dispersie simplu de calculat, alegeti dimensiunea
tabelei ca fiind 11.)

Poate aveti nevoie:

$$ASC('a') = 97$$

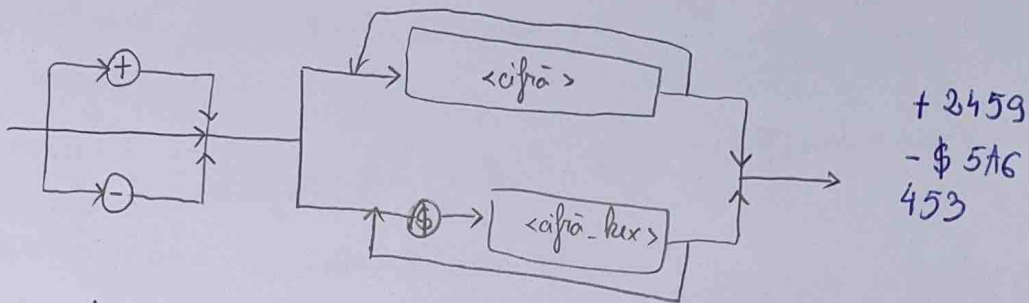
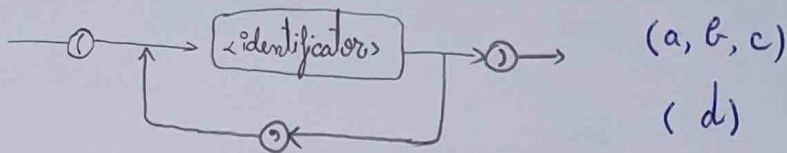
$$ASC('0') = 48$$

LFTC - SEMINAR 1

→ studiem procesul de compilare

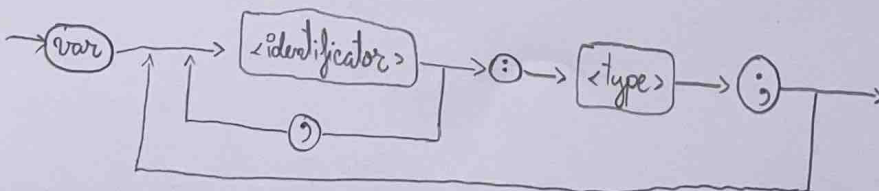
I Diagramele de sintaxă sunt repr. grafice folosite pt. a descrie sintaxa limbajelor de programare.

① Dați câte 2 exemple valide care respectă urm. specificații



② Descrieți, folosind diagrame de sintaxă, sintaxa secțiunii de declarații de variabile în Pascal. „Neterminalele” (vor apărea în dreptunghiuri cu colțuri rotunjite) vor fi <ID> și <type>.

ex: var : a, b, c : integer;
s: string;
v: array [1...20] of boolean;



2) Dați o descriere echivalentă în BNF și EBNF corespunzătoare diagramelor din secțiunea precedentă.

EBNF: " " limbaj
[] optional

{ } de o sau mai multe ori

() grupuri

ex: $\langle \text{listă-param} \rangle ::= "(" \langle \text{ID} \rangle \{ "," \langle \text{ID} \rangle \} ")"$

BNF ex: $\langle \text{listă-par} \rangle ::= (\langle \text{listă-ID} \rangle \text{ sau } \langle \text{listă-ID} \rangle ::= \langle \text{ID} \rangle , \langle \text{listă-ID} \rangle \mid \langle \text{ID} \rangle$

rezolvare:

1.1.2 BNF: $\langle \text{mr} \rangle ::= \langle \text{semnr} \rangle \langle \text{cifre} \rangle \mid \langle \text{semnr} \rangle \$ \langle \text{cifre-hexa} \rangle$
 $\langle \text{cifre} \rangle ::= \langle \text{cifră} \rangle \langle \text{cifre} \rangle \mid \langle \text{cifră} \rangle$
 $\langle \text{cifre-hexa} \rangle ::= \langle \text{cifră-hexa} \rangle \langle \text{cifre-hexa} \rangle \mid \langle \text{cifră-hexa} \rangle$
 $\langle \text{semnr} \rangle ::= + \mid - \mid \epsilon$

EBNF: $\langle \text{mr} \rangle = ["+" | "-"] (\langle \text{cifră} \rangle \{ \langle \text{cifră} \rangle \} / \$ \langle \text{cifră-hexa} \rangle \{ \langle \text{cifră-hexa} \rangle \})$

3) 1) Fie o gramatică ce descrie sintaxa unui mini-limbaj de programare, dată prin regulile de producție: ...

a) identificați terminalii și neterminalele gramaticii

TERMINALE: begin, end, ID, ;, =, +, if, then, (,), .

NETERMINALE: $\langle \text{program} \rangle$, $\langle \text{listă-imbr} \rangle$, $\langle \text{imbr} \rangle$, $\langle \text{atribuire} \rangle$, $\langle \text{imbr-if} \rangle$, $\langle \text{expr} \rangle$, $\langle \text{variabilă} \rangle$

b) Dați două "mini-programe" de scris de specificat date. Regulile de formare pentru un ID sunt: $a(a|b|c)^*$

begin
ac = a
if (a) then ac = ab + a
end

begin
abc = aa + ab + ac
ac = abc
end

② În următorul exemplu de program în Pascal:

```
var f, a1, a2, a3: integer;
```

begin

$$a_1 := \gamma;$$
$$a_2 := 11;$$
$$a^3 := a_1 + a_2 + a_3;$$
$$g := 5$$

end

a) identif. elemente lexicali și structurale sintactice

- elem lexica:

curvinte cheie: var, integer, begin, end

ID: f, a_1, a_2, a_3

val. const: 7, 11, 3, 5

operatori : $:=, +$

separator: " " , " " , " " , " "

- strukturi sintactice:

declarative

attribuire

instr. compară

ble

program

6) descrieți sintaxa structurilor sintactice folosind BNF, EBNF sau gr. independente de context

$$\langle \text{program} \rangle ::= \langle \text{declarative} \rangle \langle \text{block} \rangle$$

```
<declarative>::= var <e_id> : <tip>;
```

$$\langle l_{-id} \rangle = 10/15, \langle l_{-id} \rangle$$

<tip> ::= integer | boolean | string | char | real

$\langle \text{block} \rangle ::= \text{begin} \langle \text{list-instr} \rangle \text{end}$

$$\langle \text{lista} - \text{imetro} \rangle ::= \langle \text{imetro} \rangle / \langle \text{imetro}; \langle \text{lista} - \text{imetro} \rangle$$
$$\langle \text{im} \sigma \rangle = \langle \text{ab} \rangle$$
$$\langle \text{attribuer} \rangle ::= ID ::= \langle \text{expression} \rangle$$

$\langle \text{expression} \rangle ::= \text{CONST} \mid \text{ID} \langle \text{op} \rangle \langle \text{expression} \rangle$

$$\langle \sigma_p \rangle = +1 - 1 \times 1\% = 1\%$$

c) scrieti un program diferit de cel de mai sus care respecta descrierile date

```
var rez, a, b: integer;
```

begin

a := 6;

$$b := 15;$$
$$\text{res} := a + b * 2$$

end;

d) FIP și Tabela de simboluri (tabelă unică și constantă, ordonare lexicografică după simbol)

TABELA INTERNĂ

atom	cod atom
ID	0
CONST	1
var	2
:	3
:	4
,	5
integer	6
boolean	7
string	8
char	9
real	10
begin	11
end	12
.	13
:=	14
+	15
-	16
*	17
%	18
/	19

FIP

cod atom	cod TS
2	-
0	1
5	-
0	2
5	-
0	3
5	-
0	4
3	-
6	-
4	-
11	-
0	2
14	-
1	5
4	-
0	3
14	-
1	6
4	-
0	4
14	-
0	2
15	-
0	3
15	-
1	4
4	-
0	1
14	-
1	8
12	-
13	-

TABELĂ SIMBOLURI (lexicografică)

simbol	cod TS
11	6
3	7
5	8
7	5
a1	2
a2	3
a3	4
f	1

(trebuie să înceapă cu 0)