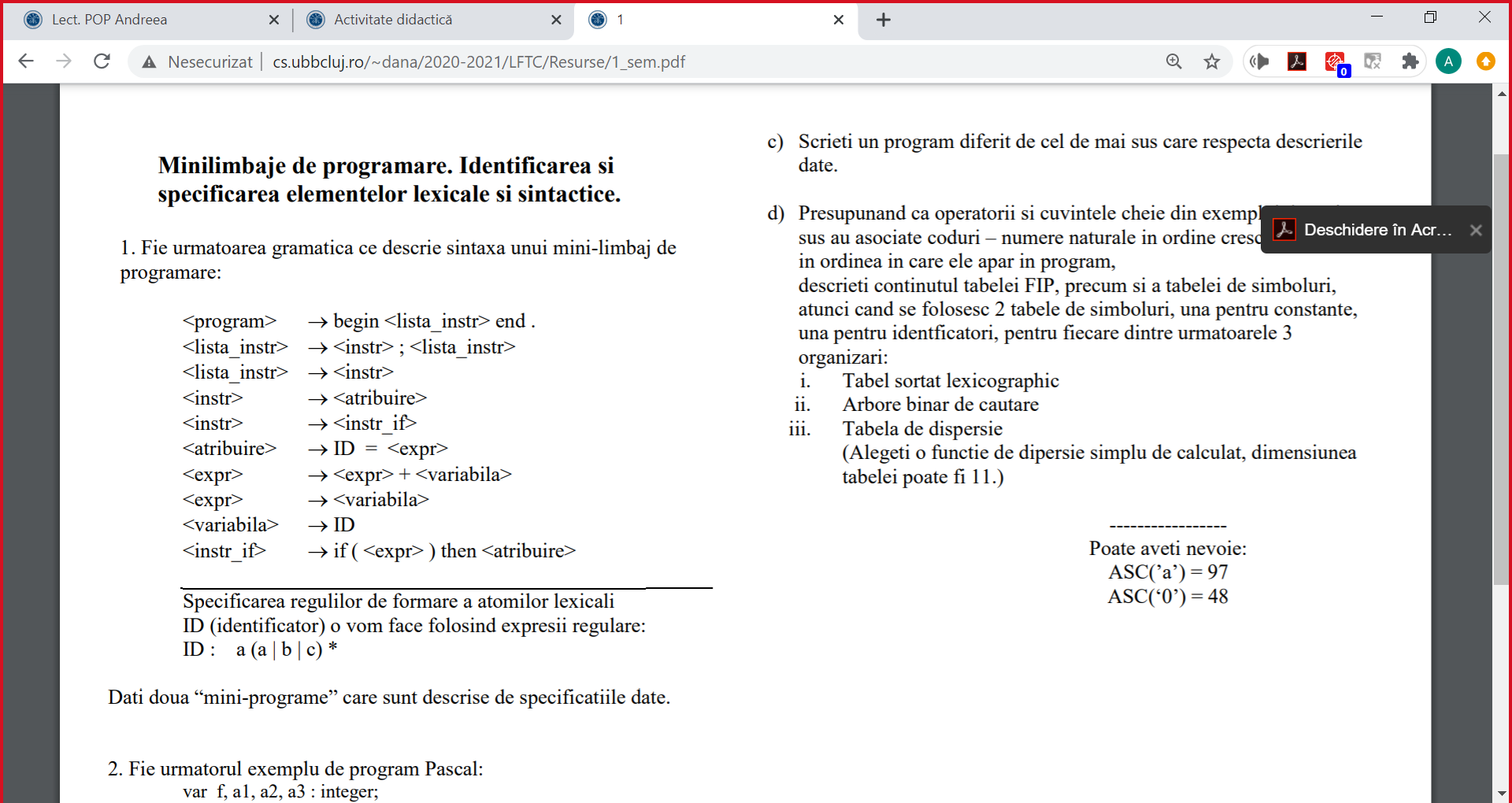
0 

<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>

ID: a ( a| b | c) \*5

Arbore de derivare:

<program>

begin

<lista\_instr>

end

.

<instr>

<atribuire>

ID

abc

=

<expr>

<variabila>

ID

abcbc

begin

abc = abcbc

end .

begin

if (abc) then

abc = abcbc

end.

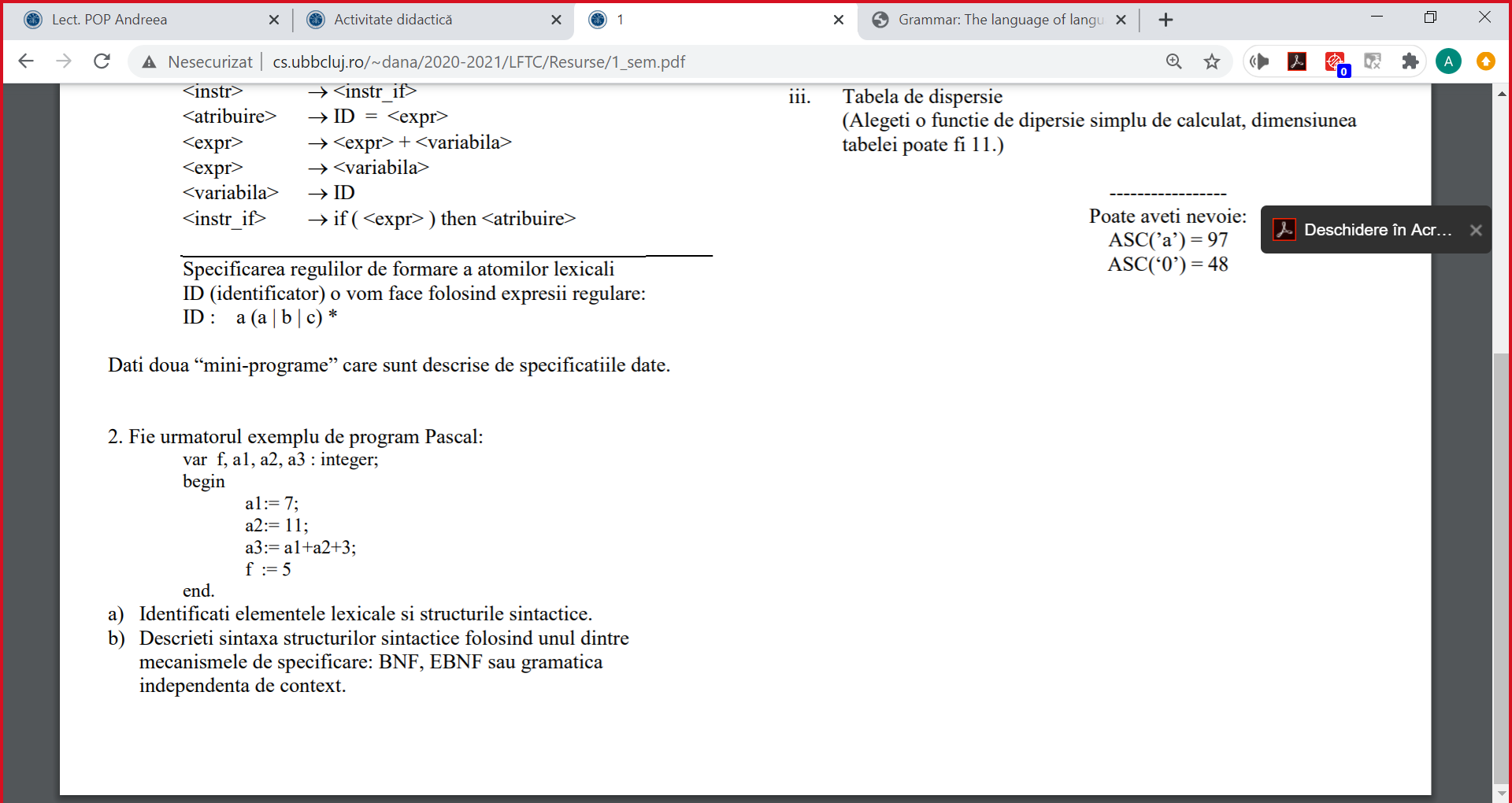
begin

abbbc = abc;

if (abbbc) then

abbbc = abc + abbbc

end.



**Elemente lexicale:**

ID: a1, a2, a3, f

Operatori: :=, +

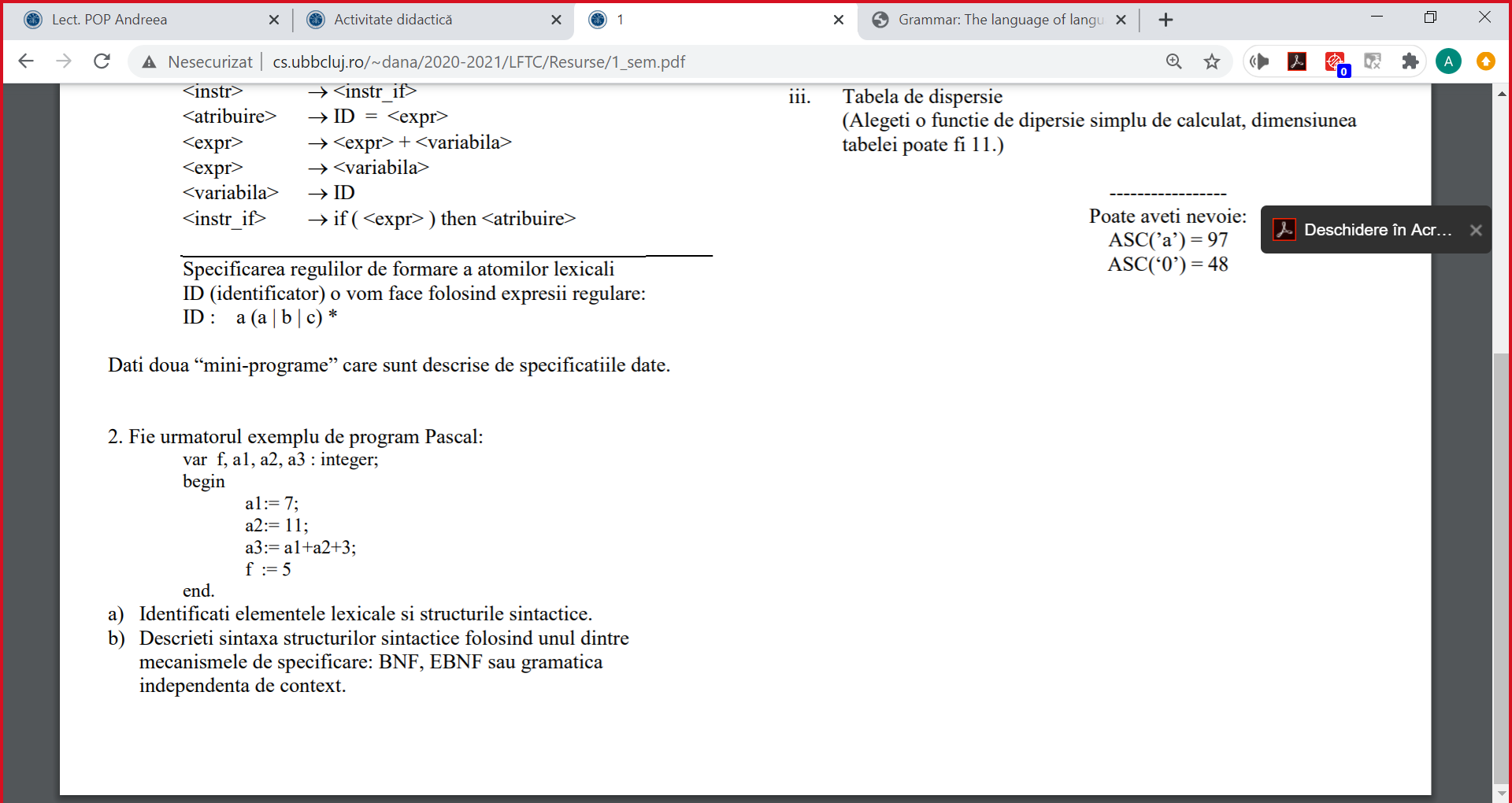
Separatori/Delimitatori: ”;”, ”,”, ”.”, ”:”

CONST: 7, 11, 3, 5

Cuvinte cheie: begin, end, var, integer

**Structurile sintactice:**

declarație, atribuire, corp+program, expresie



<program>  <declaratie> begin <lista\_instr> end**.**

<lista\_instr>  <instr> ; <lista\_instr>

<lista\_instr>  <instr>

<instr>  <atribuire>

<atribuire>ID := <expr>

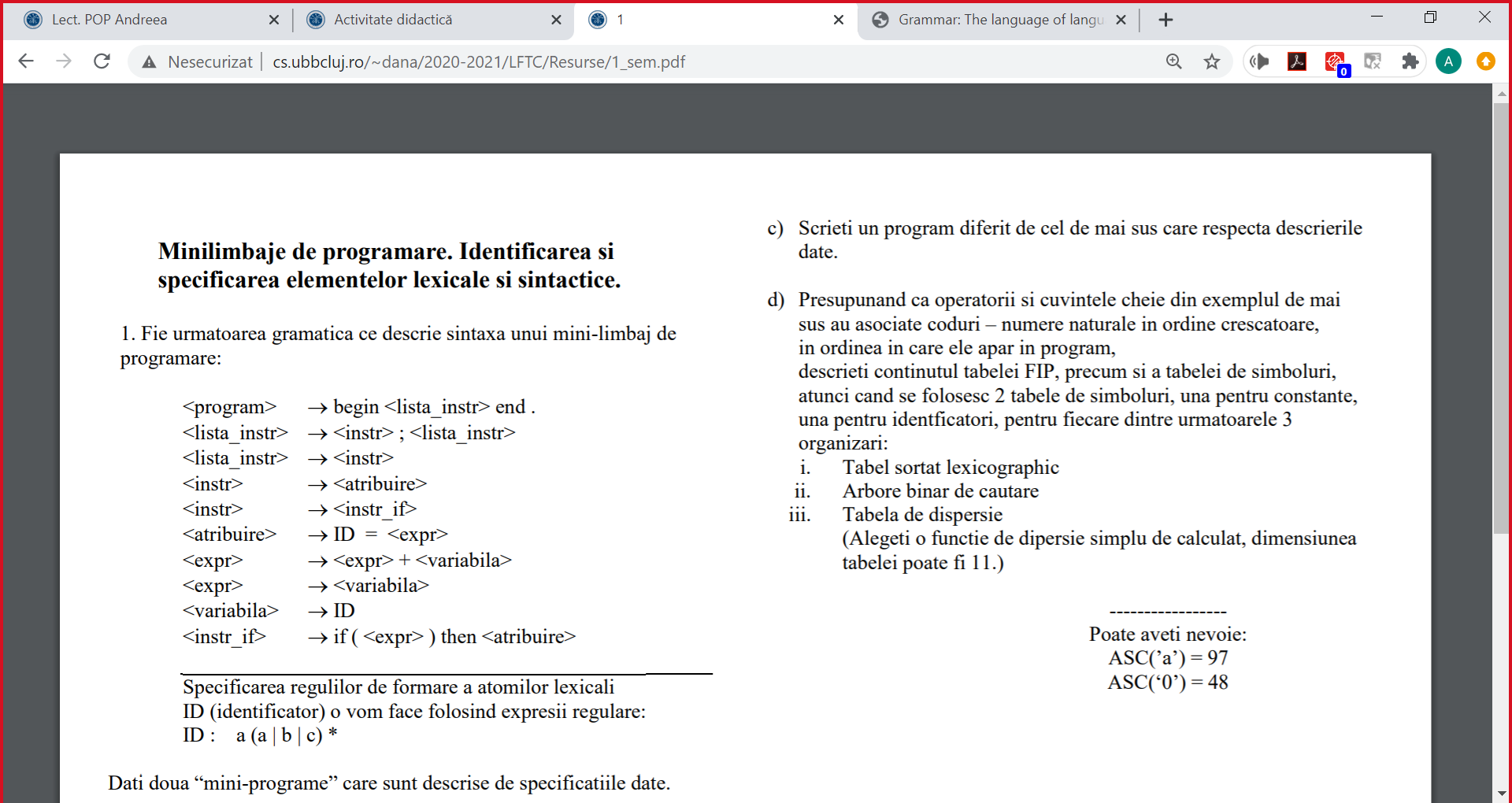
<expr>  <expr> + <termen>

<expr>  <termen>

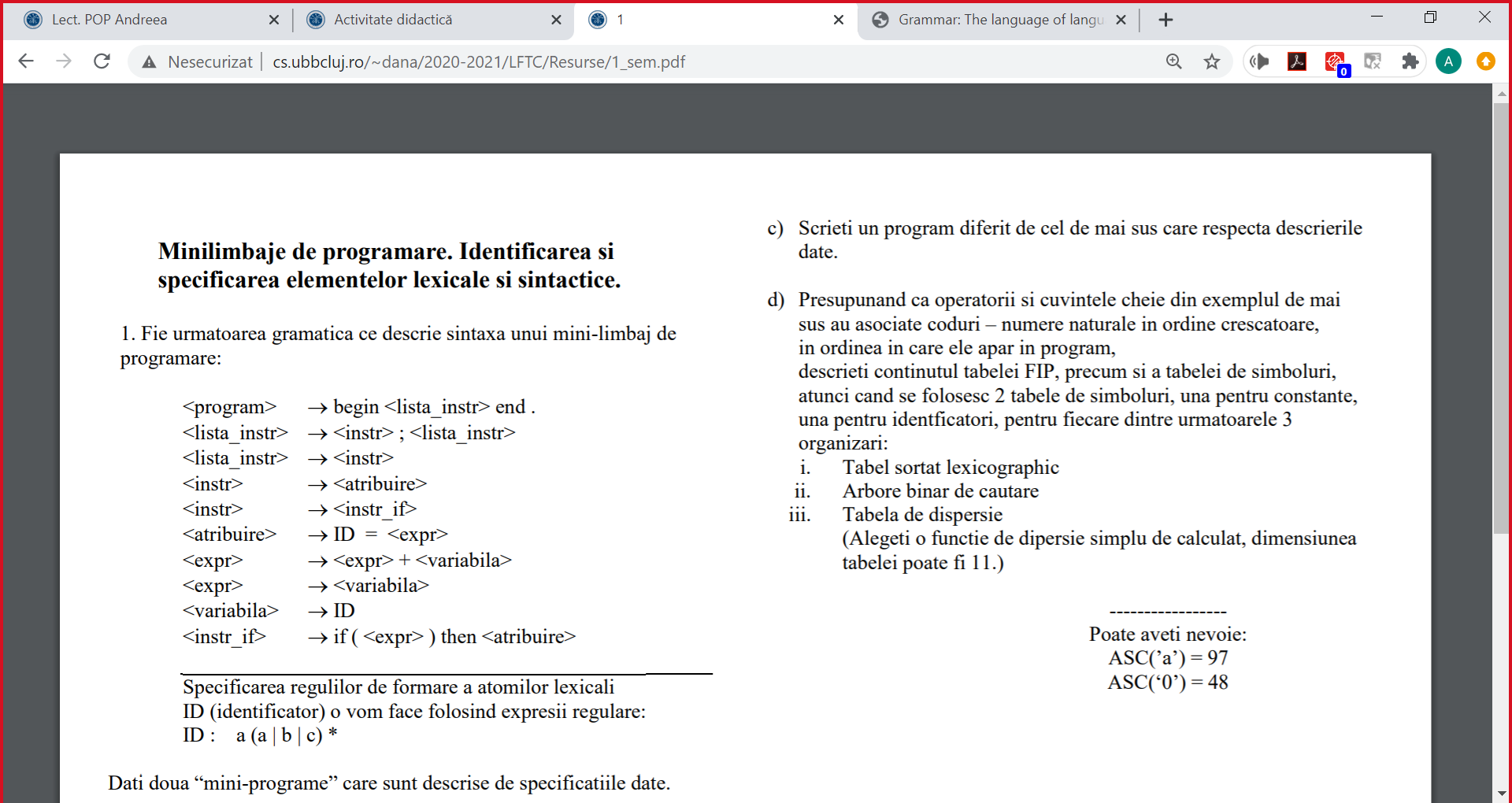
<termen>  ID | CONST

<declaratie>  var <lista\_variabile> : integer;

<lista\_variabile>  ID | ID, <lista\_variabile>



|  |  |
| --- | --- |
| var a,b,c : integer;  begin  a:=2;  b:=3;  c:=a;  a:=b;  b:=c  end. | <program>  <declaratie> begin <lista\_instr> end**.**  <lista\_instr>  <instr> ; <lista\_instr>  <lista\_instr>  <instr>  <instr>  <atribuire>  <atribuire>ID := <expr>  <expr>  <expr> + <termen>  <expr>  <termen>  <termen> -> ID | CONST  <declaratie> -> var <lista\_variabile> : integer;  <lista\_variabile> -> ID | ID, <lista\_variabile> |



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| tabelă ordonată lexicofgrafic (alphabetic) | | | | | | | |
| **Atom lexical** | **Cod Atom** | **Programul (fis. de intrare)** | FIP  Forma Internă a Programului | | TSA  Tabela de Simboluri (ID) | | |
| ***Cod Atom*** | ***COD TS*** | Simbol (ID) | Cod TS | |
| a1  a2  a3  f | 1  2  3  0 | |
| ID  CONST  begin  end  var  .  integer:  :=  ,  +  ; | 0  1  2  3  4  5  6  7  8  9  10  11 | var  f  ,  a1  ,  a2  ,  a3  :  integer  ;  begin  a1  :=  7  ;  a2  :=  11  ;  a3  :=  a1  +  a2  +  3  ;  f  :=  5  end  . | 4  0  9  0  9  0  9  0  7  6  11  2  0  8  1  11  0  8  1  11  0  8  0  10  0  10  1  11  0  8  1  3  5 | -  0  -  1  -  2  -  3  -  -  -  -  1  -  0  -  2  -  1  -  3  -  1  -  2  -  2  -  0  -  3  -  - | TS  Tabela de Simboluri (CONST) | | |
| Simbol (CONST) | | Cod TS |
| 11  3  5  7 | | 1  2  3  0 |

Arbore binar de căutare

f (0)

a1 (1)

a2 (2)

a3 (3)

7 (0)

11 (1)

3 (2)

5 (3)

Tabelă de dispersie unică ID+ CONST

|  |  |
| --- | --- |
| **Cod hash** | **ID/CONST** |
| 0 | 7 |
| 1 |  |
| 2 |  |
| 3 | f |
| 4 | a1 |
| 5 | a2 |
| 6 | a3 |
| 7 | 3 |
| 8 |  |
| 9 | 5 |
| 10 | 11 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| tabelă de dispersie cu dim. 11  funcția de dispersie = suma ASCII char. mod 11 | | | | | | | |
| **Atom lexical** | **Cod Atom** | **Programul (fis. De intrare)** | FIP  Forma Internă a Programului | | TS vezi mai sus | | |
| ***Cod Atom*** | ***COD TS*** |  |  | |
|  |  | |
| ID  CONSTbegin  end  var  .  integer:  :=  ,  +  ; | 0  1  2  3  4  5  6  7  8  9  10  11 | var  f  ,  a1  ,  a2  ,  a3  :  integer  ;  begin  a1  :=  7  ;  a2  :=  11  ;  a3  :=  a1  +  a2  +  3  ;  f  :=  5  end  . | 4  0  9  0  9  0  9  0  7  6  11  2  0  8  1  11  0  8  1  11  0  8  0  10  0  10  1  11  0  8  1  3  5 | -  3  -  4  -  5  -  6  -  -  -  -  4  -  0  -  5  -  10  -  6  -  4  -  5  -  7  -  3  -  9  -  - |  | | |
|  | |  |
|  | |  |