Project: Write semantics for the rules given in Table 1 such that the Subset Pascal Compiler that prints identifiers (**id**) and lists of identifiers (*identifier_list*) with actual identifiers extracted from the Subset Pascal program that was compiled. Write the identifiers and lists of identifiers in the trace file. Enclose lists of strings in braces separated by commas. Parenthesize identifiers.

Also, print and parenthesize the strings that represented by the tokens **intlit**, **realit**, and **chrlit**.

A sample program is given in Figure 1 and the corresponding trace file is shown in Figure 2.

			Rule
No.	**LHS**		**RHS**
7	*program_parameter_list*	→	*identifier_list*
8	*identifier_list*	→	**id**
9	*identifier_list*	→	*identifier_list* **, id**
10	*declarations*	→	∈
11	*declarations*	→	*declarations* **var** *identifier_list* **:** *type* **;**
14	*standard_type*	→	**id**
18	*subprogram_head*	→	**function id** *subprogram_parameters* **:** *standard_type* **;**
19	*subprogram_head*	→	**procedure id** *subprogram_parameters* **;**
22	*parameter_list*	→	*identifier_list* **:** *type*
23	*parameter_list*	→	*parameter_list* **;** *identifier_list* **:** *type*
34	*variable*	→	**id**
35	*variable*	→	**id [** *expression* **]**
36	*procedure_statement*	→	**id**
37	*procedure_statement*	→	**id (** *expression_list* **)**

Table 1. Subset Pascal Grammar (continued)

		Rule	
No.	**LHS**		**RHS**
63	*factor*	→	**id**
64	*factor*	→	**id [** *expression* **]**
65	*factor*	→	**id (** *expression_list* **)**
68	*factor*	→	**intlit**
69	*factor*	→	**realit**
70	*factor*	→	**chrlit**

Table 1. Subset Pascal Grammar (continued)

```
program p01;
    var a,b,c,d:integer;
begin
end.
```

Figure 1. Subset Pascal Program p01.

```
Token:Code=293 Name=   PROGRAM line=  1 col=  1 Spelling="program"
Token:Code=300 Name=        ID line=  1 col=  9 Spelling="p01"
Token:Code=265 Name= SEMICOLON line=  1 col= 12 Spelling=";"
program_parameters -> empty
declarations -> empty
Token:Code=297 Name=       VAR line=  2 col=  3 Spelling="var"
Token:Code=300 Name=        ID line=  2 col=  7 Spelling="a"
identifier_list -> ID(a)
Token:Code=264 Name=     COMMA line=  2 col=  8 Spelling=","
Token:Code=300 Name=        ID line=  2 col=  9 Spelling="b"
identifier_list -> {a} , ID(b)
Token:Code=264 Name=     COMMA line=  2 col= 10 Spelling=","
Token:Code=300 Name=        ID line=  2 col= 11 Spelling="c"
identifier_list -> {a,b} , ID(c)
Token:Code=264 Name=     COMMA line=  2 col= 12 Spelling=","
Token:Code=300 Name=        ID line=  2 col= 13 Spelling="d"
identifier_list -> {a,b,c} , ID(d)
Token:Code=266 Name=     COLON line=  2 col= 14 Spelling=":"
Token:Code=300 Name=        ID line=  2 col= 15 Spelling="integer"
standard_type -> ID(integer)
type -> standard_type
Token:Code=265 Name= SEMICOLON line=  2 col= 22 Spelling=";"
declarations -> declarations VAR identifier_list {a,b,c,d}: type;
Token:Code=281 Name=     BEGIN line=  3 col=  1 Spelling="begin"
```

Figure 2. Trace file for Subset Pascal Program p01

```
subprogram_declarations -> empty
Token:Code=285 Name=       END line=  4 col=  1 Spelling="end"
optional_statements -> empty
compound_statement -> BEGIN optional_statement END
Token:Code=263 Name=    PERIOD line=  4 col=  4 Spelling="."
program   ->   PROGRAM   ID(p01)   program_parameters   ;   declarations
subprogram_declarations compound_statement .
```

Figure 2. Trace file for Subset Pascal Program p01 (continued)