1. Based on the grammar of the μ*C* language listed below, please draw the parse tree for the input. (**Note:** 1) Please use the leftmost derivation method to derive the rules. A derivation that always chooses the leftmost possible nonterminal at each step is called a leftmost derivation. 2) When performing the derivation to generate the Input Program, the reserved words, i.e., int and double, have the higher priority for token (Terminal) matching. For example, when the "int" string is occurred, the string will only match the INT token rather than ID token.) (3pt)

Input Program

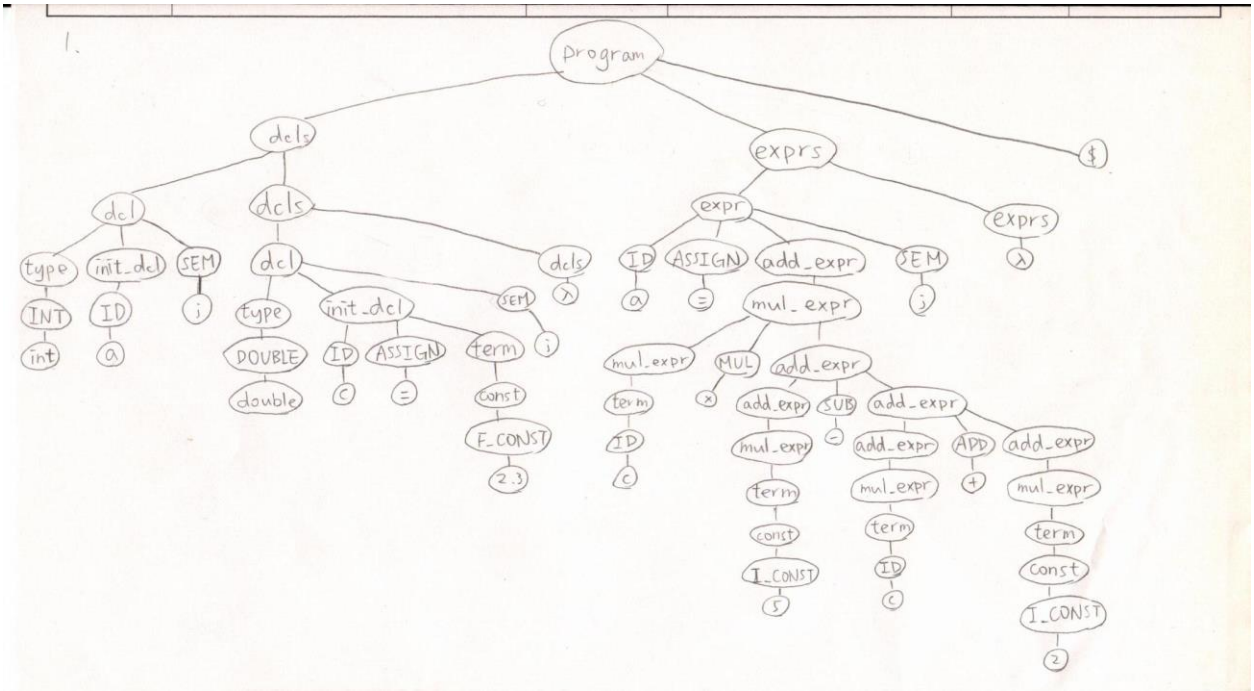
```
int a;
double c = 2.3;

a = c * 5 - c + 2;
```

Grammar of μC language

```
program   → dcls exprs $
exprs     → expr exprs
          | λ
dcls      → dcl dcls
          | λ
expr      → ID ASSIGN add_expr SEM
add_expr  → mul_expr
          | add_expr ADD add_expr
          | add_expr SUB add_expr
mul_expr  → term
          | mul_expr MUL add_expr
          | mul_expr DIV add_expr
dcl       → type init_dcl SEM
type      → INT
          | DOUBLE
init_dcl  → ID ASSIGN term
          | ID
term      → ID
          | const
const     → I_CONST
          | F_CONST
```

Terminal	Regular Expression
INT	"int"
DOUBLE	"double"
ASSIGN	"="
SEM	";"
ADD	"+"
SUB	"-"
MUL	"*"
DIV	"/"
I_CONST	$[0-9]^+$
F_CONST	$[0-9]^+.[0-9]^+$
ID	[a-zA-Z][a-zA-Z0-9]*

ANS:



2. Most languages are case sensitive, so keywords can be written only one way, and the regular expressions describing their lexeme is very simple. However, some languages, like SQL, are **case insensitive**, so a keyword can be written either in lowercase or in uppercase, or in any mixture of cases; for example, the SQL keyword CREATE can also be written create, Create, or cReAtE. Please show the regular expression of the keyword, **create**, for SQL. (2pt)

ANS:

The case insensitive regular expression of “create” is [Cc][Rr][Ee][Aa][Tt][Ee].