

## Θεμα 2

i)

**A.M. Φοιτητή 1:** ΜΠΠΛ2107**9**

**A.M. Φοιτητή 2:** ΜΠΠΛ2106**2**

**Δήλωση μεταβλητών:**

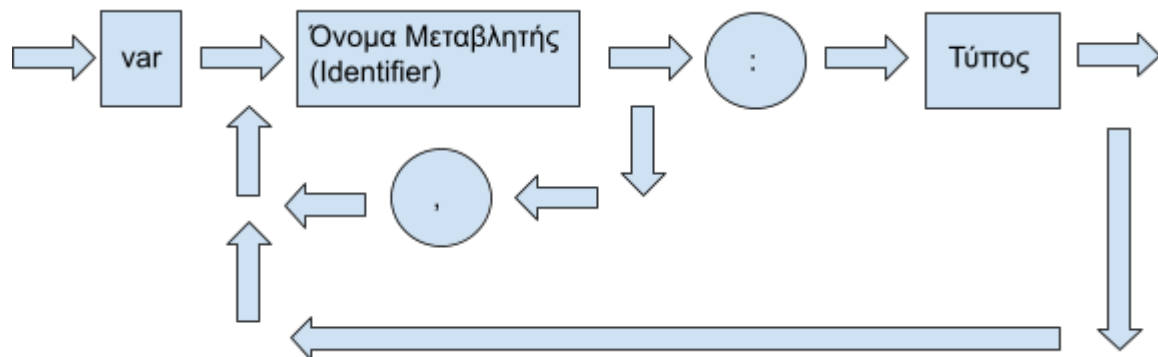
var count, sum: **integer****92**

average: **real****92**

is\_member: **boolean****92**

fullname: **char****92**

**Συνακτικό Διάγραμμα:**



ii)

Usage	Notation
definition	=
concatenation	,
termination	;
alternation	
optional	[ ... ]
repetition	{ ... }

```
letter = "A" | "B" | "C" | "D" | "E" | "F" | "G"  
        | "H" | "I" | "J" | "K" | "L" | "M" | "N"  
        | "O" | "P" | "Q" | "R" | "S" | "T" | "U"  
        | "V" | "W" | "X" | "Y" | "Z" | "a" | "b"  
        | "c" | "d" | "e" | "f" | "g" | "h" | "i"  
        | "j" | "k" | "l" | "m" | "n" | "o" | "p"  
        | "q" | "r" | "s" | "t" | "u" | "v" | "w"  
        | "x" | "y" | "z" ;
```

```
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;  
symbol = "," | ":" | "[" | "]" | "{" | "}" | ";" ;  
identifier = letter [{letter | digit | "_" | "-"}] ;  
var_type = integer92 | real92 | boolean92 | char92 ;  
var {identifier [{, identifier}] : var_type} ;
```

iii)

Το πρόγραμμά μας σε lex βρίσκεται στο αρχείο **exercise2\_3.l**. Επίσης, έχουμε το αρχείο **text.txt** με δηλώσεις μεταβλητών που αναγνωρίζονται από τον λεκτικό αναλυτή μας. Το import του text.txt στον λεκτικό αναλυτή μας, γίνεται με εντολή, κατά το τρέξιμο του εκτελέσιμου και υπάρχουν οδηγίες στην αρχή του **exercise2\_3.l** σε comments για όλα τα στάδια εκτέλεσης.

iv)

Αρχικά το πρώτο πρόβλημα του ορισμού της γλώσσας είναι ότι δεν ορίζει αυστηρά το τέλος εντολής. Για να αποφύγουμε αυτό το πρόβλημα θα μπορούσαμε να θέσουμε ως κανόνα το ότι χρειάζεται κάποιο σύμβολο, όπως π.χ. το ερωτηματικό μετά το τέλος κάθε εντολής. Αν θέταμε, όμως, τον παραπάνω κανόνα θα έπρεπε το var να επαναλαμβάνεται στην αρχή κάθε εντολής, άρα και μετά την πρώτη δηλώση των μεταβλητών μας.