

---

# Συμβολισμοί Ενδιάμεσου Κώδικα

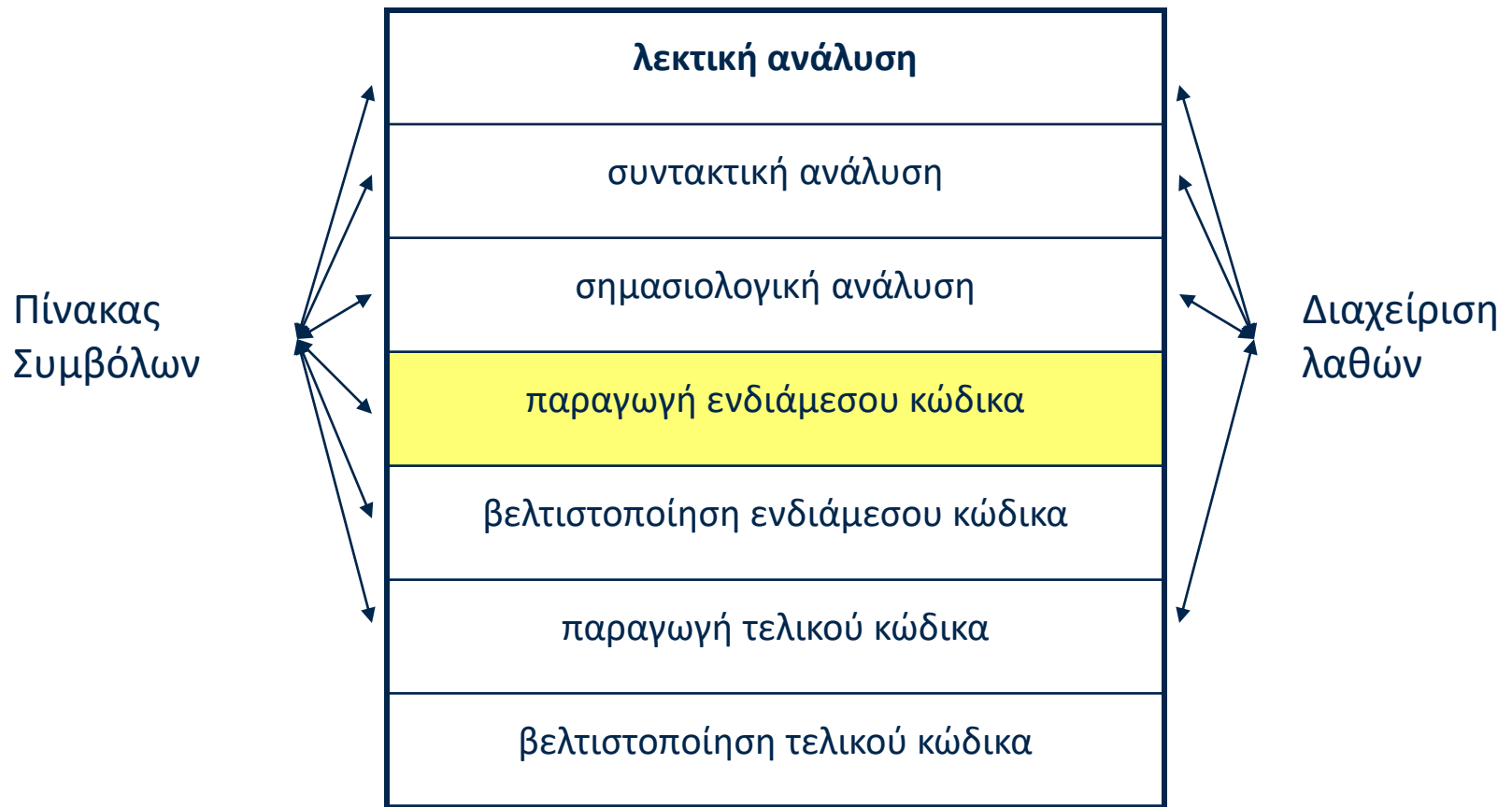
Διαλέξεις στο μάθημα: Μεταφραστές  
Γεώργιος Μανής

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
UNIVERSITY OF IOANNINA



## Οι Φάσεις της Μεταγλώττισης

---



## *Παραγωγή Ενδιάμεσου Κώδικα*

---



## *Παραγωγή Ενδιάμεσου Κώδικα*

---



## Ενδιάμεση Γλώσσα

---

- ‡ Ο ενδιάμεσος κώδικας είναι ένα σύνολο από τετράδες

- ένας τελεστής
- τρία τελούμενα

π.χ. + , a,b,t\_1

\*, t\_1,2,t\_2

:=,t\_2,\_,c

- ‡ Οι τετράδες είναι αριθμημένες. Κάθε τετράδα έχει μπροστά της έναν μοναδικό αριθμό που τη χαρακτηρίζει. Μόλις τελειώσει η εκτέλεση μίας τετράδας εκτελείται η τετράδα που έχει τον αμέσως μεγαλύτερο αριθμό, εκτός εάν η τετράδα που μόλις εκτελέστηκε υποδείξει κάτι διαφορετικό .

π.χ.: 100: +,a,b,c

110: +,d,e,f

## Οι τελεστές αριθμητικών πράξεων

---

τετράδες της μορφής:

$op, x, y, z$

- όπου το  $op$  μπορεί να είναι ένα εκ των:  $+, -, *, /$
- τα τελούμενα  $x, y$  μπορεί να είναι:
  - ονόματα μεταβλητών
  - αριθμητικές σταθερές
- το τελούμενο  $z$  μπορεί να είναι:
  - όνομα μεταβλητής

## Οι τελεστές αριθμητικών πράξεων

---

τετράδες της μορφής:

op, x, y, z

- εφαρμόζεται ο τελεστής op στα τελούμενα x και y και το αποτέλεσμα τοποθετείται στο τελούμενο z

π.χ.:

+ , a, b, c      αντιστοιχεί στην πράξη  $c = a + b$

/ , a, b, c      αντιστοιχεί στην πράξη  $c = a / b$

## Ο τελεστής εκχώρησης

---

τετράδες της μορφής:

$:=, x, \_, z$

- το τελούμενο  $x$  μπορεί να είναι:
  - όνομα μεταβλητής
  - αριθμητική σταθερά
- το τελούμενο  $z$  μπορεί να είναι:
  - όνομα μεταβλητής
- η τιμή του  $x$  εκχωρείται στη μεταβλητή  $z$ 
  - αντιστοιχεί στη εκχώρηση  $z:=x$



## Παράδειγμα

---

r:=4

pi:=3.14

area = pi \* r \* r

θα μπορούσε ισοδύναμα  
να είχαμε:



χωρίς να σημαίνει ότι  
αυτό ακριβώς θα βγάλει  
και ο μεταγλωττιστής

100: :=,4,\_,r

110: :=,3.14,\_,pi

120: \*,pi,r,T\_1

130: \*,T\_1,r,area

## Τελεστής άλματος χωρίς συνθήκη

---

jump, \_, \_, z

μεταπήδηση χωρίς όρους στη θέση z

π.χ. 100: :=,1,\_x

110: jump 130

120: :=,2,\_x

130 ...

όταν φτάσουμε στο 130 η τιμή του x θα είναι 1 και όχι 2

## Τελεστής άλματος με συνθήκη

---

relop, x, y, z

όπου relop ένας από τους τελεστές

=, >, <, <>, >=, <=

μεταπήδηση στη θέση z αν ισχύει η x relop y

π.χ. 100: =,a,4,120

110: jump,\_,\_,140

120: :=,1,\_b

130: jump 150

140: :=,2,\_b

150: ...

το b θα έχει την τιμή 1 αν ισχύει η συνθήκη a=4 και 2 αν δεν ισχύει

---

## *Αρχή και τέλος ενότητας*

---

- ‡ `begin_block, name, _, _`  
αρχή υποπρογράμματος ή προγράμματος με το όνομα `name`
  - ‡ `end_block, name, _, _`  
τέλος υποπρογράμματος ή προγράμματος με το όνομα `name`
  - ‡ `halt, _, _, _`  
τερματισμός προγράμματος
-

## *Αρχή και τέλος ενότητας*

---

Παράδειγμα begin\_block - end\_block, name - halt

100: begin\_block, add, \_, \_

110: := , 1 , \_ , x

120: := , 2 , \_ , y

130: + , x , y , z

140: halt , \_ , \_ , \_

150: end\_block , add , \_ , \_

---

## Συναρτήσεις - Διαδικασίες

---

# par, x, m, \_

όπου x παράμετρος συνάρτησης και m ο τρόπος μετάδοσης

CV : μετάδοση με τιμή

REF: μετάδοση με αναφορά

RET: επιστροφή τιμής συνάρτησης

# call, name, \_, \_

κλήση συνάρτησης name

# ret, x, \_, \_

επιστροφή τιμής συνάρτησης


---

## Παράδειγμα κλήσης συνάρτησης

---

`x := foo (in a, inout b)`

θα μπορούσε ισοδύναμα  
να είχαμε:



χωρίς να σημαίνει ότι  
αυτό ακριβώς θα βγάλει  
και ο μεταγλωττιστής

```
100: par , a , cv , _  
110: par , b , ref , _  
120: par , T_1 , ret , _  
130: call , foo , _ , _ , _  
140: ... τιμή στο x
```


## Παράδειγμα κλήσης διαδικασίας

---

call foo (in a, inout b)

θα μπορούσε ισοδύναμα  
να είχαμε:

χωρίς να σημαίνει ότι  
αυτό ακριβώς θα βγάλει  
και ο μεταγλωττιστής



```
100: par , a , cv , _  
110: par , b , ref , _  
120: call , foo , _ , _ , _  
130 ...
```



## Βοηθητικές Υπορουτίνες

---

- # nextquad()
  - επιστρέφει τον αριθμό της επόμενης τετράδας που πρόκειται να παραχθεί
- # genquad(op, x, y, z)
  - δημιουργεί την επόμενη τετράδα (op, x, y, z)
- # newtemp()
  - δημιουργεί και επιστρέφει μία νέα προσωρινή μεταβλητή
  - οι προσωρινές μεταβλητές είναι της μορφής  
T\_1, T\_2, T\_3 ...

## Βοηθητικές Υπορουτίνες

---

- # `emptylist()`
  - δημιουργεί μία κενή λίστα ετικετών τετράδων
- # `makelist(x)`
  - δημιουργεί μία λίστα ετικετών τετράδων που περιέχει μόνο το  $x$
- # `mergelist(list1, list2)`
  - δημιουργεί μία λίστα ετικετών τετράδων από τη συνένωση των λιστών  $list_1$ ,  $list_2$
- # `backpatch(list,z)`
  - η λίστα  $list$  αποτελείται από δείκτες σε τετράδες των οποίων το τελευταίο τελούμενο δεν είναι συμπληρωμένο
  - η `backpatch` επισκέπτεται μία μία τις τετράδες αυτές και τις συμπληρώνει με την ετικέτα  $z$