

Ενδιάμεσος κώδικας

(i)

- Λόγοι ύπαρξης
 - Διευκολύνει το έργο της μετάφρασης
 - Διευκολύνει τη βελτιστοποίηση
 - Διευκολύνει την κατάτμηση σε εμπρόσθιο και οπίσθιο τμήμα

Ενδιάμεσος κώδικας

(ii)

- Μετάφραση οδηγούμενη από τη σύνταξη (syntax-directed translation)
 - Για κάθε δομή της γλώσσας προσδιορίζεται ο αντίστοιχος ενδιάμεσος κώδικας
 - Διευρύνεται ο συντακτικός αναλυτής με σημασιολογικές ρουτίνες που παράγουν ενδιάμεσο κώδικα
- Σχέδιο παραγωγής ενδιάμεσου κώδικα
- Μεταβλητές ιδιοτήτων (attributes) για κάθε σύμβολο της γραμματικής

Ενδιάμεση γλώσσα

(i)

- Τετράδες (quadruples)

$n: op, x, y, z$

- Παράδειγμα: $b*b-4*a*c$
 - 1: *, b, b, \$1
 - 2: *, 4, a, \$2
 - 3: *, \$2, c, \$3
 - 4: -, \$1, \$3, \$4

Ενδιάμεση γλώσσα

(ii)

- Τριάδες (triples)

$n: op, x, y$

- Παράδειγμα: $b*b-4*a*c$

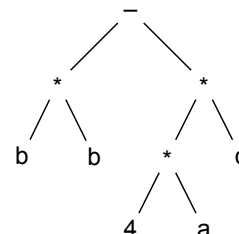
- 1: *, b, b
- 2: *, 4, a
- 3: *, (2), c
- 4: -, (1), (3)

Ενδιάμεση γλώσσα

(iii)

- Αφηρημένα συντακτικά δέντρα (abstract syntax trees)

- Παράδειγμα: $b*b-4*a*c$



Ενδιάμεση γλώσσα

(iv)

- Προθεματικός και επιθεματικός κώδικας (prefix/postfix code)

- Παράδειγμα: $b*b-4*a*c$

- * b b * * 4 a c προθεματικός
b b * 4 a * c * - επιθεματικός

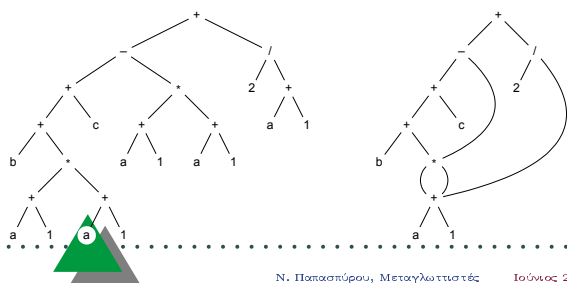
Ενδιάμεση γλώσσα

(v)

- Κατευθυνόμενοι ακυκλικοί γράφοι (directed acyclic graphs)

- Παράδειγμα:

```
b+(a+1)*(a+1)+c-(a+1)*(a+1)+2/(a+1)
```



Γλώσσα τετράδων

- Μορφή τετράδας:

$n: op, x, y, z$

όπου:

- n : ετικέτα τετράδας (φυσικός αριθμός)
 - op : τελεστής
 - x, y, z : τελούμενα
- Ανάλογα με το είδος του τελεστή, κάποια τελούμενα ενδεχομένως παραλείπονται

Τελούμενα

(i)

- Σταθερά
 - ακέραια, πραγματική, λογική
 - χαρακτήρας, συμβολοσειρά, nil
- Όνομα
 - μεταβλητή, παραμέτρος, υποπρόγραμμα
- Προσωρινή μεταβλητή: $\$n$
- Αποτέλεσμα συνάρτησης: $\$\$$
- Αποδεικτοδότηση: $[x]$ x απλό τελούμενο
- Διεύθυνση: $\{x\}$ x απλό τελούμενο

Τελούμενα

(ii)

- Ετικέτα
 - εντολής στο αρχικό πρόγραμμα
 - τετράδας
- Τρόπος περάσματος
 - V : κατ' αξία
 - R : κατ' αναφορά
 - RET : θέση αποτελέσματος συνάρτησης
- Κενό : -
- Προσωρινά κενό : * (για backpatching)

Τελεστές

(i)

- unit, $I, -, -$
- endu, $I, -, -$
αρχή και τέλος δομικής μονάδας
- op, x, y, z $op \in \{+, -, *, /, \%\}$
 $z := x op y$
- $:=, x, -, z$
 $z := x$
- array, x, y, z
 $z :=$ η διεύθυνση του στοιχείου $x[y]$

Τελεστές

(ii)

- op, x, y, z $op \in \{=, <, >, <=, >=\}$
αν $x op y$ τότε πήγαινε στην τετράδα z
- ifb, $x, -, z$
αν η λογική τιμή x είναι αληθής τότε πήγαινε στην τετράδα z
- jump, $-, -, z$
πήγαινε στην τετράδα z
- label, $I, -, -$
jump1, $-, -, I$
ορισμός ετικέτας και άλμα προς αυτήν

Τελεστές

(iii)

- **call**, $-, -, I$
κάλεσε τη δομική μονάδα I
- **par**, $x, m, -$
πέρασε την πραγματική παράμετρο x με τρόπο περάσματος m
- **ret**, $-, -, -$
επιστροφή από την τρέχουσα δομική μονάδα

Μεταβλητές ιδιοτήτων

- **PLACE**: θέση όπου βρίσκεται αποθηκευμένη η τιμή μιας l-value ή μιας r-value
- **TYPE**: τύπος μιας l-value ή μιας r-value
- **NEXT**: λίστα από ετικέτες τετράδων που περιέχουν άλματα στην επόμενη εντολή
- **TRUE, FALSE**: λίστες από ετικέτες τετράδων που περιέχουν άλματα στον κώδικα που πρέπει να εκτελεστεί αν μια συνθήκη είναι αληθής ή ψευδής

Βοηθητικές υπορουτίνες (i)

- **NEXTQUAD()**
Επιστρέφει τον αριθμό της επόμενης τετράδας
- **GENQUAD**(op, x, y, z)
Γεννά την επόμενη τετράδα op, x, y, z
- **NEWTEMP**(t)
Δημιουργεί μια νέα προσωρινή μεταβλητή τύπου t
- **EMPTYLIST()**
Δημιουργεί μια κενή λίστα ετικετών τετράδων

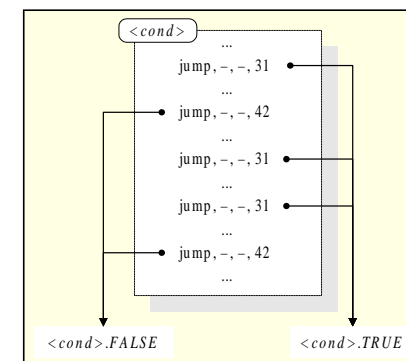
Βοηθητικές υπορουτίνες (ii)

- **MAKELIST**(x)
Δημιουργεί μια λίστα ετικετών τετράδων που περιέχει μόνο το στοιχείο x
- **MERGE**(l_1, \dots, l_n)
Συνένωση των λιστών ετικετών τετράδων $l_1 \dots l_n$
- **BACKPATCH**(l, z)
Αντικαθιστά σε όλες τις τετράδες που περιέχονται στην l την άγνωστη ετικέτα τετράδας με τη z (**backpatching**)

Αριθμητικές εκφράσεις

- **Ακέραιες σταθερές**
 $\langle r\text{-value} \rangle ::= \langle \text{integer-const} \rangle \{ P_1 \}$
 $P_1 : \{ \langle r\text{-value} \rangle.PLACE = \langle \text{integer-const} \rangle; \}$
- **Τελεστές με δύο τελούμενα**
 $\langle r\text{-value} \rangle ::= \langle \text{expr} \rangle \langle \text{binop} \rangle \langle \text{expr} \rangle \{ P_{14} \}$
 $P_{14} : \{ W = \text{NEWTEMP}(\langle r\text{-value} \rangle.TYPE);$
 $\text{GENQUAD}(\langle \text{binop} \rangle.NAME,$
 $\langle \text{expr} \rangle^1.PLACE,$
 $\langle \text{expr} \rangle^2.PLACE, W);$
 $\langle r\text{-value} \rangle.PLACE = W; \}$

Λογικές εκφράσεις (i)

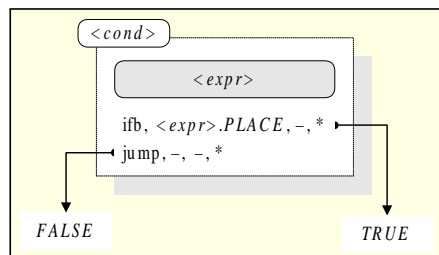


Λογικές εκφράσεις

(ii)

- Λογικές εκφράσεις σε συμβολισμό 0/1

$\langle \text{cond} \rangle ::= \langle \text{expr} \rangle$



Λογικές εκφράσεις

(iii)

- Λογικές εκφράσεις σε συμβολισμό 0/1

$\langle \text{cond} \rangle ::= \langle \text{expr} \rangle \{ P_{21} \}$

$P_{21} : \{ \langle \text{cond} \rangle . \text{TRUE} = \text{MAKELIST}(\text{NEXTQUAD}());$
 $\text{GENQUAD}(\text{ifb}, \langle \text{expr} \rangle . \text{PLACE}, -, *);$
 $\langle \text{cond} \rangle . \text{FALSE} = \text{MAKELIST}(\text{NEXTQUAD}());$
 $\text{GENQUAD}(\text{jump}, -, -, *); \}$

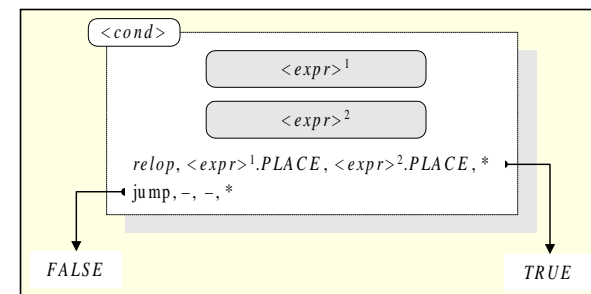


Λογικές εκφράσεις

(iv)

- Τελεστές σύγκρισης

$\langle \text{cond} \rangle ::= \langle \text{expr} \rangle^1 \langle \text{relop} \rangle \langle \text{expr} \rangle^2$



Λογικές εκφράσεις

(v)

- Τελεστές σύγκρισης

$\langle \text{cond} \rangle ::= \langle \text{expr} \rangle^1 \langle \text{relop} \rangle \langle \text{expr} \rangle^2 \{ P_{23} \}$

$P_{23} : \{ \langle \text{cond} \rangle . \text{TRUE} = \text{MAKELIST}(\text{NEXTQUAD}());$
 $\text{GENQUAD}(\langle \text{relop} \rangle . \text{NAME},$
 $\langle \text{expr} \rangle^1 . \text{PLACE},$
 $\langle \text{expr} \rangle^2 . \text{PLACE}, *);$
 $\langle \text{cond} \rangle . \text{FALSE} = \text{MAKELIST}(\text{NEXTQUAD}());$
 $\text{GENQUAD}(\text{jump}, -, -, *); \}$

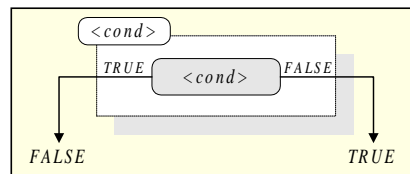


Λογικές εκφράσεις

(vi)

- Άρνηση

$\langle \text{cond} \rangle ::= \text{"not"} \langle \text{cond} \rangle$

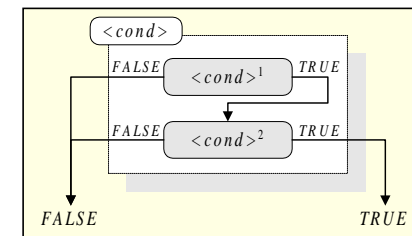


Λογικές εκφράσεις

(vii)

- Σύζευξη

$\langle \text{cond} \rangle ::= \langle \text{cond} \rangle^1 \text{"and"} \langle \text{cond} \rangle^2$



Λογικές εκφράσεις

(viii)

■ Σύζευξη

$\langle \text{cond} \rangle ::= \langle \text{cond} \rangle_1 \text{ "and" } \{ P_{25} \} \langle \text{cond} \rangle_2 \{ P_{26} \}$
 $P_{25} : \{ \text{BACKPATCH}(\langle \text{cond} \rangle_1. \text{TRUE}, \text{NEXTQUAD}()); \}$
 $P_{26} : \{ \langle \text{cond} \rangle. \text{FALSE} = \text{MERGE}(\langle \text{cond} \rangle_1. \text{FALSE}, \langle \text{cond} \rangle_2. \text{FALSE}); \}$
 $\langle \text{cond} \rangle. \text{TRUE} = \langle \text{cond} \rangle_2. \text{TRUE}; \}$

Απλές εντολές

■ Κενή εντολή

$\langle \text{stmt} \rangle ::= \epsilon \{ P_{29} \}$
 $P_{29} : \{ \langle \text{stmt} \rangle. \text{NEXT} = \text{EMPTYLIST}(); \}$

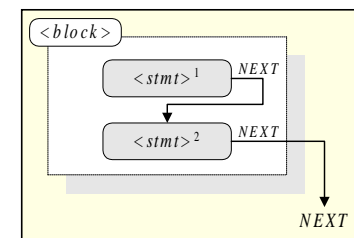
■ Εντολή ανάθεσης

$\langle \text{stmt} \rangle ::= \langle \text{l-value} \rangle \text{ "!=" } \langle \text{expr} \rangle \{ P_{30} \}$
 $P_{30} : \{ \text{GENQUAD}(\text{"!="}, \langle \text{expr} \rangle. \text{PLACE}, -, \langle \text{l-value} \rangle. \text{PLACE}); \}$
 $\langle \text{stmt} \rangle. \text{NEXT} = \text{EMPTYLIST}(); \}$

Σύνθετη εντολή

(i)

$\langle \text{stmt} \rangle ::= \langle \text{block} \rangle$
 $\langle \text{block} \rangle ::= \text{"begin"} \langle \text{stmt} \rangle (\text{";" } \langle \text{stmt} \rangle)^* \text{"end"}$



Σύνθετη εντολή

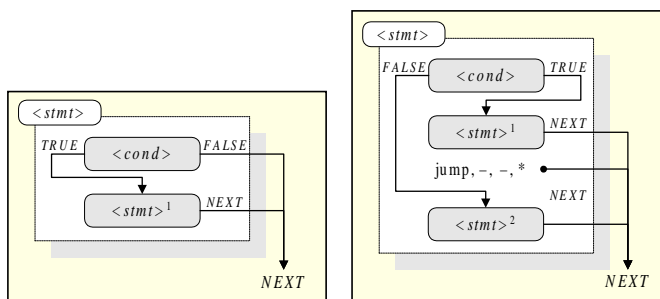
(ii)

$\langle \text{stmt} \rangle ::= \langle \text{block} \rangle \{ P_{34} \}$
 $P_{34} : \{ \langle \text{stmt} \rangle. \text{NEXT} = \langle \text{block} \rangle. \text{NEXT}; \}$
 $\langle \text{block} \rangle ::= \text{"begin"} \langle \text{stmt} \rangle^1 \{ P_{35} \}$
 $(\text{";" } \{ P_{36} \} \langle \text{stmt} \rangle^2 \{ P_{37} \})^* \text{"end"} \{ P_{38} \}$
 $P_{35} : \{ L = \langle \text{stmt} \rangle^1. \text{NEXT}; \}$
 $P_{36} : \{ \text{BACKPATCH}(L, \text{NEXTQUAD}()); \}$
 $P_{37} : \{ L = \langle \text{stmt} \rangle^2. \text{NEXT}; \}$
 $P_{38} : \{ \langle \text{block} \rangle. \text{NEXT} = L; \}$

Εντολή if

(i)

$\langle \text{stmt} \rangle ::= \text{"if"} \langle \text{cond} \rangle \text{"then"} \langle \text{stmt} \rangle [\text{"else"} \langle \text{stmt} \rangle]$



Εντολή if

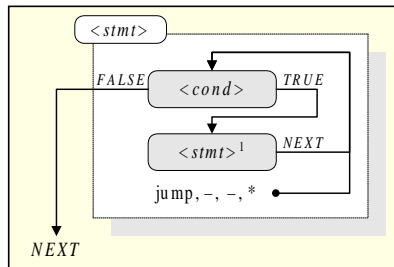
(ii)

$\langle \text{stmt} \rangle ::= \text{"if"} \langle \text{cond} \rangle \{ P_{39} \} \text{"then"} \langle \text{stmt} \rangle^1$
 $[\text{"else"} \{ P_{40} \} \langle \text{stmt} \rangle^2 \{ P_{41} \}] \{ P_{42} \}$
 $P_{39} : \{ \text{BACKPATCH}(\langle \text{cond} \rangle. \text{TRUE}, \text{NEXTQUAD}());$
 $L_1 = \langle \text{cond} \rangle. \text{FALSE};$
 $L_2 = \text{EMPTYLIST}(); \}$
 $P_{40} : \{ L_1 = \text{MAKELIST}(\text{NEXTQUAD}());$
 $\text{GENQUAD}(\text{jump}, -, -, *);$
 $\text{BACKPATCH}(\langle \text{cond} \rangle. \text{FALSE}, \text{NEXTQUAD}()); \}$
 $P_{41} : \{ L_2 = \langle \text{stmt} \rangle^2. \text{NEXT}; \}$
 $P_{42} : \{ \langle \text{stmt} \rangle. \text{NEXT} = \text{MERGE}(L_1, \langle \text{stmt} \rangle^1. \text{NEXT}, L_2); \}$

Εντολή while

(i)

$\langle \text{stmt} \rangle ::= \text{"while"} \langle \text{cond} \rangle \text{"do"} \langle \text{stmt} \rangle$



Εντολή while

(ii)

$\langle \text{stmt} \rangle ::= \text{"while"} \{ P_{43} \} \langle \text{cond} \rangle \text{"do"} \{ P_{44} \} \langle \text{stmt} \rangle^1 \{ P_{45} \}$

$P_{43} : \{ Q = \text{NEXTQUAD}(); \}$

$P_{44} : \{ \text{BACKPATCH}(\langle \text{cond} \rangle.\text{TRUE}, \text{NEXTQUAD}()); \}$

$P_{45} : \{ \text{BACKPATCH}(\langle \text{stmt} \rangle^1.\text{NEXT}, Q); \}$
 $\text{GENQUAD}(\text{jump}, -, -, Q);$
 $\langle \text{stmt} \rangle.\text{NEXT} = \langle \text{cond} \rangle.\text{FALSE}; \}$

Κλήση υποπρογραμμάτων (i)

$\langle \text{call} \rangle ::= \langle \text{id} \rangle \text{"("} [\langle \text{expr} \rangle (\text{","} \langle \text{expr} \rangle)^*] \text{"}"}$

$\langle \text{r-value} \rangle ::= \langle \text{call} \rangle$

$\langle \text{stmt} \rangle ::= \langle \text{call} \rangle$

- Πέρασμα παραμέτρων με τετράδες par
- Πέρασμα θέσης αποτελέσματος με τετράδα par αν πρόκειται για συνάρτηση
- Κλήση με τετράδα call

Κλήση υποπρογραμμάτων (ii)

$\langle \text{call} \rangle ::= \langle \text{id} \rangle \text{"("} \{ P_{46} \} [\langle \text{expr} \rangle^1 \{ P_{47} \} (\text{","} \langle \text{expr} \rangle^2 \{ P_{48} \})^*] \text{"}"} \{ P_{49} \}$

$P_{46} : \{ N = 1; \}$

$P_{47} : \{ \text{GENQUAD}(\text{"par"}, \langle \text{expr} \rangle^1.\text{PLACE}, \text{PARAMMODE}(\langle \text{id} \rangle, N), -);$
 $N = N + 1; \}$

$P_{48} : \{ \text{GENQUAD}(\text{"par"}, \langle \text{expr} \rangle^2.\text{PLACE}, \text{PARAMMODE}(\langle \text{id} \rangle, N), -);$
 $N = N + 1; \}$

Κλήση υποπρογραμμάτων (iii)

$\langle \text{call} \rangle ::= \langle \text{id} \rangle \text{"("} \{ P_{46} \} [\langle \text{expr} \rangle^1 \{ P_{47} \} (\text{","} \langle \text{expr} \rangle^2 \{ P_{48} \})^*] \text{"}"} \{ P_{49} \}$

(συνέχεια)

$P_{49} : \{ \text{if} (\text{ISFUNCTION}(\langle \text{id} \rangle)) \{$
 $W = \text{NEWTEMP}(\text{FUNCRESULT}(\langle \text{id} \rangle));$
 $\text{GENQUAD}(\text{par}, \text{RET}, W, -);$
 $\langle \text{call} \rangle.\text{PLACE} = W;$
 $\}$
 $\text{GENQUAD}(\text{call}, -, -, \langle \text{id} \rangle); \}$

Κλήση υποπρογραμμάτων (iv)

- Κλήση συνάρτησης

$\langle \text{r-value} \rangle ::= \langle \text{call} \rangle \{ P_{50} \}$

$P_{50} : \{ \langle \text{r-value} \rangle.\text{PLACE} = \langle \text{call} \rangle.\text{PLACE}; \}$

- Κλήση διαδικασίας

$\langle \text{stmt} \rangle ::= \langle \text{call} \rangle \{ P_{51} \}$

$P_{51} : \{ \langle \text{stmt} \rangle.\text{NEXT} = \text{EMPTYLIST}(); \}$

Κλήση υποπρογραμμάτων (v)

- Επιστροφή από υποπρόγραμμα

$\langle \text{stmt} \rangle ::= \text{"return"} [\langle \text{expr} \rangle \{ P_{52} \}] \{ P_{53} \}$

$P_{52} : \{ \text{GENQUAD}(\text{retv}, \langle \text{expr} \rangle . \text{PLACE}, -, -); \}$

$P_{53} : \{ \text{GENQUAD}(\text{ret}, -, -, -); \}$

- Δήλωση υποπρογράμματος

$\langle \text{body} \rangle ::= (\langle \text{local} \rangle)^* \{ P_{56} \} \langle \text{block} \rangle \text{";" } \{ P_{57} \}$

$P_{56} : \{ \text{GENQUAD}(\text{unit}, I, -, -); \}$

$P_{57} : \{ \text{BACKPATCH}(\langle \text{block} \rangle . \text{NEXT}, \text{NEXTQUAD}());$
 $\text{GENQUAD}(\text{endu}, I, -, -); \}$