

Αρχές Γλωσσών Προγραμματισμού

Χρήστος Νομικός

Τμήμα Μηχανικών Η/Υ και Πληροφορικής
Πανεπιστήμιο Ιωαννίνων

2015

- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Σύμβολα, Αλφάβητα και Συντακτικές Δομές

Αλφάβητο ονομάζεται ένα πεπερασμένο σύνολο συμβόλων.

Συμβολίζουμε με Σ^* το σύνολο όλων των πεπερασμένων ακολουθιών από σύμβολα που ανήκουν Σ .

Παράδειγμα 1. Αν $\Sigma = \{a, b, c\}$ τότε

$$abacab \in \Sigma^*, a \in \Sigma^*, daa \notin \Sigma^*$$

Παράδειγμα 2. Αν $\Sigma = \{+, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ τότε

$$-90876 \in \Sigma^*, 023+45-+ \in \Sigma^*, 0.345 \notin \Sigma^*$$

Σύμβολα, Αλφάβητα και Συντακτικές Δομές

Ονομάζουμε γλώσσα ή συντακτική δόμη από το αλφάβητο Σ ένα υποσύνολο του Σ^* .

Στη συνέχεια θα προτιμηθεί ο όρος συντακτική δομή, για να μη γίνεται σύγχυση με τη γλώσσα προγραμματισμού.

Παράδειγμα 3. Αν $\Sigma = \{a, b, c\}$ τότε το σύνολο των συμβολοσειρών που αποτελούνται από n συνεχόμενα a ακολουθούμενα από n συνεχόμενα b είναι μία συντακτική δομή από το Σ .

Παράδειγμα 4. Αν $\Sigma = \{+, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ τότε το σύνολο των συμβολοσειρών που παριστάνουν προσημασμένους ακέραιους είναι μία συντακτική δομή από το Σ .

Οι συντακτικές δομές αποτελούν σύνολα συμβολακολουθιών και άρα μπορούν να περιγραφούν όπως ένα οποιοδήποτε σύνολο. Συνήθως όμως επιλέγεται ένας από τους παρακάτω τρόπους:

- Περιγράφεται ένας συστηματικός τρόπος παραγωγής όλων των συμβολοσειρών της συντακτικής δομής. Αυτό μπορεί να γίνει χρησιμοποιώντας π.χ. γραμματικές (σύνολα κανόνων παραγωγής), συμβολισμό BNF ή συντακτικά διαγράμματα.
- Δίνεται ένας συστηματικός τρόπος αναγνώρισης των συμβολοσειρών που ανήκουν στη συντακτική δομή. Αυτό γίνεται περιγράφοντας μία μηχανή ή έναν αλγόριθμο που αποφασίζει αν μία συμβολοσειρά ανήκει ή όχι στη γλώσσα.

Ο πρώτος τρόπος περιγραφής είναι πιο κατάλληλος για να περιγράψουμε ή να κατανοήσουμε τη σύνταξη μιας γλώσσας προγραμματισμού. Ο δεύτερος είναι απαραίτητος κατά την υλοποίηση της γλώσσας, αφού απαιτείται μεταξύ άλλων συντακτική ανάλυση του προγράμματος.

Σύμβολα, Αλφάβητα και Συντακτικές Δομές

Το αλφάβητο που χρησιμοποιεί μια γλώσσα προγραμματισμού κατά κανόνα περιλαμβάνει:

- Τα γράμματα του λατινικού αλφαβήτου a-z, A-Z.
- Τα αριθμητικά ψηφία 0-9.
- Άλλα σύμβολα, όπως
= + - * / () , . & ' ' ! < > %

Ορισμένες γλώσσες προγραμματισμού διαχωρίζουν τα κεφαλαία από τα μικρά γράμματα και ονομάζονται case sensitive (π.χ. C, Java, Haskell, Prolog, Python).

Σε άλλες γλώσσες προγραμματισμού τα κεφαλαία γράμματα είναι ισοδύναμα με τα αντίστοιχα μικρά (π.χ. Pascal).

Σύμβολα, Αλφάβητα και Συντακτικές Δομές

Τα στοιχεία του αλφαβήτου χρησιμοποιούνται για τη δημιουργία λεκτικών μονάδων (π.χ. αριθμητικών σταθερών, αναγνωριστικών που χρησιμοποιούνται για ονόματα μεταβλητών, τύπων ή συναρτήσεων, τελεστών όπως είναι το \leq ή \geq , κλπ)

Οι παραπάνω λεκτικές μονάδες χρησιμοποιούνται για το σχηματισμό του προγράμματος.

Ο παραπάνω χωρισμός της σύνταξης σε δύο επίπεδα αν και δεν είναι απαραίτητος για την περιγραφή της σύνταξης, είναι εξαιρετικά χρήσιμος κατά τη συντακτική ανάλυση.

Ορισμένες λέξεις έχουν συγκεκριμένη σημασία σε μια γλώσσα προγραμματισμού και δεν μπορούν να χρησιμοποιηθούν από τον προγραμματιστή για άλλο σκοπό. Οι λέξεις αυτές ονομάζονται δεσμευμένες λέξεις.

Για παράδειγμα οι δεσμευμένες λέξεις στην Pascal είναι:

and	array	begin	case	const
div	do	downto	else	end
file	for	function	goto	if
in	label	mod	nil	not
of	or	packed	procedure	program
record	repeat	set	then	to
type	until	var	while	with

- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)**
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Περιγραφή σύνταξης με BNF

Ο συμβολισμός BNF είναι μία μεταγλώσσα με την οποία μπορούμε να περιγράψουμε τη σύνταξη μιας συντακτικής δομής (γλώσσας).

Η περιγραφή της σύνταξης μίας συντακτικής δομής σε BNF αποτελείται από ένα σύνολο κανόνων.

Κάθε κανόνας περιγράφει τη κατασκευή μίας σύνθετης συντακτικής δομής, από άλλες συντακτικές δομές και σύμβολα του αλφαβήτου.

Οι συντακτικές δομές παριστάνονται με συντακτικές μεταβλητές που ονομάζονται και μη τερματικά σύμβολα. Μια συντακτική μεταβλητή γράφεται μέσα σε \langle και \rangle , ώστε να διαχωρίζεται από τα τερματικά σύμβολα (στοιχεία του αλφαβήτου).

Ένας κανόνας BNF έχει τη μορφή

$$V ::= X_{1,1} \dots X_{1,k_1} \mid \dots \mid X_{n,1} \dots X_{1,k_n}$$

όπου

- $n > 0$ και $k_1, k_2, \dots, k_n \geq 0$.
- V είναι μη τερματικό σύμβολο.
- $X_{i,j}$ είναι μη τερματικό σύμβολο ή σύμβολο του αλφαβήτου.
- $::=$ και \mid είναι μετασύμβολα του BNF (όπως και τα $<$, $>$).

Ο παραπάνω κανόνας δηλώνει ότι η συντακτική δομή που αντιστοιχεί στη μεταβλητή V μπορεί να έχει n εναλλακτικές μορφές (διαχωρίζονται με \mid). Η i -στη εναλλακτική μορφή συμπεριλαμβάνει όλες τις συμβολοσειρές που προκύπτουν από την παράθεση k_i συμβολοσειρών.

Περιγραφή σύνταξης με BNF

Για κάθε μή τερματικό σύμβολο Y που εμφανίζεται δεξιά του $::=$ σε κάποιον κανόνα, θα πρέπει να υπάρχει ένας κανόνας με κεφαλή το Y . Με άλλα λόγια θα πρέπει να καθορίζεται κάθε συντακτική δομή που εμφανίζεται στους κανόνες.

Αν $k_i = 0$ τότε το V μπορεί να είναι η κενή ακολουθία συμβόλων. Σε αυτή την περίπτωση χρησιμοποιούμε το σύμβολο ϵ .

Για να περιγράψουμε μια συντακτική δομή που προκύπτει επαναλαμβάνοντας μία άλλη δομή ένα οποιοδήποτε πλήθος φορές, χρησιμοποιούμε αναδρομή.

Περιγραφή σύνταξης με BNF

Στη συνέχεια περιγράφουμε το πώς καθορίζεται το σύνολο των συμβολοσειρών που αντιστοιχεί σε ένα μη τερματικό σύμβολο με βάση ένα σύνολο κανόνων S .

Έστω S ένα σύνολο κανόνων BNF στο οποίο το σύνολο των τερματικών συμβόλων είναι Σ και το σύνολο των μή τερματικών συμβόλων είναι V .

Αν α και β είναι ακολουθίες από τερματικά και μη τερματικά σύμβολα (δηλαδή $\alpha, \beta \in (\Sigma \cup V)^*$), γράφουμε $\alpha \Rightarrow_S \beta$ για να δηλώσουμε ότι η ακολουθία β παράγεται από την α , αντικαθιστώντας μία εμφάνιση ενός μή τερματικού συμβόλου με κάποια από τις εναλλακτικές μορφές που μπορεί να πάρει με βάση τους κανόνες του S .

Το $\alpha \Rightarrow_S^n \beta$ δηλώνει το ότι το β παράγεται από την α σε n βήματα και μπορεί να οριστεί αναδρομικά:

- $\alpha \Rightarrow_S^0 \beta$ αν $\beta = \alpha$,
- $\alpha \Rightarrow_S^{n+1} \beta$ αν υπάρχει γ τέτοιο ώστε $\alpha \Rightarrow_S^n \gamma$ και $\gamma \Rightarrow_S \beta$

Το $\alpha \Rightarrow_S^* \beta$ δηλώνει το ότι το β παράγεται από την α σε πεπερασμένο αριθμό βημάτων, δηλαδή ότι υπάρχει k τέτοιο ώστε $\alpha \Rightarrow_S^k \beta$

Περιγραφή σύνταξης με BNF

Το σύνολο των συμβολοσειρών που αντιστοιχεί σε ένα μη τερματικό σύμβολο $\langle x \rangle$ με βάση ένα σύνολο κανόνων S είναι το σύνολο των ακολουθιών τερματικών συμβόλων που παράγονται από το $\langle x \rangle$, δηλαδή το σύνολο

$$\{w \in \Sigma^* \mid \langle x \rangle \Rightarrow_S^* w\}$$

Μπορούμε να παραλείψουμε το δείκτη S όταν είναι ξεκάθαρο σε ποιο σύνολο κανόνων BNF αναφερόμαστε.

Παρότι σε μία παραγωγή μπορούμε να αντικαθιστούμε οποιοδήποτε μη τερματικό σύμβολο θέλουμε, θα αρκούσε να αντικαθιστούμε κάθε φορά το αριστερότερο μη τερματικό σύμβολο. Για λόγους ομοιομορφίας θα επιλέξουμε αυτή τη στρατηγική στα παρακάτω παραδείγματα.

Παράδειγμα 5. Οι παρακάτω κανόνες σε BNF περιγράφουν τη σύνταξη των προσημασμένων ακεραίων:

$$\langle \text{ακέραιος} \rangle ::= \langle \text{πρόσημο} \rangle \langle \text{ακολουθία} \rangle$$
$$\langle \text{πρόσημο} \rangle ::= + \mid -$$
$$\langle \text{ακολουθία} \rangle ::= \langle \text{ψηφίο} \rangle \mid \langle \text{ψηφίο} \rangle \langle \text{ακολουθία} \rangle$$
$$\langle \text{ψηφίο} \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \mid 0$$

Περιγραφή σύνταξης με BNF

Για παράδειγμα, ο αριθμός -192 παράγεται με τον παρακάτω τρόπο:

<ακέραιος> \Rightarrow

<πρόσημο><ακολουθία> \Rightarrow

-<ακολουθία> \Rightarrow

-<ψηφίο><ακολουθία> \Rightarrow

-1<ακολουθία> \Rightarrow

-1<ψηφίο><ακολουθία> \Rightarrow

-19<ακολουθία> \Rightarrow

-19<ψηφίο> \Rightarrow

-192

Παράδειγμα 6. Έστω ότι στο προηγούμενο παράδειγμα θέλουμε

- το πρόσημο να είναι προαιρετικό
- ο αριθμός 0 να μην δέχεται πρόσημο
- το πρώτο ψηφίο ενός αριθμού να μην είναι το '0' εκτός αν ο αριθμός είναι το 0

Περιγραφή σύνταξης με BNF

Η σύνταξη των αριθμών περιγράφεται από τους παρακάτω κανόνες:

$\langle \text{ακέραιος} \rangle ::= 0 \mid \langle \text{θετικός} \rangle \mid \langle \text{πρόσημο} \rangle \langle \text{θετικός} \rangle$

$\langle \text{πρόσημο} \rangle ::= + \mid -$

$\langle \text{θετικός} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \mid \langle \text{ψηφίο-όχι-0} \rangle \langle \text{ακολουθία} \rangle$

$\langle \text{ακολουθία} \rangle ::= \langle \text{ψηφίο} \rangle \mid \langle \text{ψηφίο} \rangle \langle \text{ακολουθία} \rangle$

$\langle \text{ψηφίο-όχι-0} \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\langle \text{ψηφίο} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \mid 0$

Περιγραφή σύνταξης με BNF

Οι παραπάνω κανόνες μπορούν να απλουστευθούν με χρήση της κενής συμβολοσειράς ϵ :

$$\langle \text{ακέραιος} \rangle ::= 0 \mid \langle \text{πρόσημο} \rangle \langle \text{θετικός} \rangle$$
$$\langle \text{πρόσημο} \rangle ::= + \mid - \mid \epsilon$$
$$\langle \text{θετικός} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \langle \text{ακολουθία} \rangle$$
$$\langle \text{ακολουθία} \rangle ::= \epsilon \mid \langle \text{ψηφίο} \rangle \langle \text{ακολουθία} \rangle$$
$$\langle \text{ψηφίο-όχι-0} \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$
$$\langle \text{ψηφίο} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \mid 0$$

Περιγραφή σύνταξης με BNF

Για παράδειγμα, με βάση το δεύτερο σύνολο κανόνων ο αριθμός 68 παράγεται με τον παρακάτω τρόπο:

$\langle \text{ακέραιος} \rangle \Rightarrow$

$\langle \text{πρόσημο} \rangle \langle \text{θετικός} \rangle \Rightarrow$

$\langle \text{θετικός} \rangle \Rightarrow$

$\langle \text{ψηφίο-όχι-0} \rangle \langle \text{ακολουθία} \rangle \Rightarrow$

$6 \langle \text{ακολουθία} \rangle \Rightarrow$

$6 \langle \text{ψηφίο} \rangle \langle \text{ακολουθία} \rangle \Rightarrow$

$6 \langle \text{ψηφίο-όχι-0} \rangle \langle \text{ακολουθία} \rangle \Rightarrow$

$68 \langle \text{ακολουθία} \rangle \Rightarrow$

68

Μερικές παρατηρήσεις:

- Υπάρχουν περισσότεροι από ένας τρόποι (σύνολα από κανόνες) για να περιγράψουμε την ίδια γλώσσα.
- Χρησιμοποιούμε ονόματα μεταβλητών που σχετίζονται με τη φυσική σημασία της συντακτικής οντότητας που περιγράφουμε. Π.χ <ακέραιος>, <πρόσημο> αντί για <στοιχείο1>, <στοιχείο2> ή <X>, <Υ>
- Ένα συμβολικό όνομα μπορεί σε διαφορετικά σύνολα κανόνων, να αντιστοιχεί σε διαφορετικές συντακτικές δομές.
- Η σειρά των κανόνων δεν παίζει ρόλο. Όμως συνήθως προτιμάται μία από τις παρακάτω προσεγγίσεις
 - Ξεκινάμε περιγράφοντας τις πιο σύνθετες συντακτικές δομές και προχωράμε προς τις πιο απλές (top-down)
 - Ξεκινάμε περιγράφοντας τις πιο απλές συντακτικές δομές και προχωράμε προς τις πιο σύνθετες (bottom-up)

Μερικά συνηθισμένα λάθη

- Οι αναδρομικοί κανόνες θα πρέπει να περιέχουν πάντοτε μία περίπτωση στην οποία η αναδρομή σταματάει. Ο παρακάτω κανόνας δεν παράγει ποτέ ένα συντακτικό αντικείμενο που αποτελείται μόνο από τερματικά σύμβολα.

$$\langle \text{απειρο} \rangle ::= 0 \langle \text{απειρο} \rangle$$

- Αν κάποιο μη τερματικό σύμβολο εμφανίζεται μόνο στο δεξιά του $::=$ τότε το σύνολο κανόνων είναι ελλιπές.
- Αν κάποιο μη τερματικό σύμβολο, εκτός από αυτό που ορίζει τη γλώσσα, εμφανίζεται μόνο αριστερά του $::=$ τότε ο αντίστοιχος κανόνας μπορεί να παραληφθεί.

Περιγραφή σύνταξης με BNF

- Η παράθεση έχει μεγαλύτερη προτεραιότητα από το |. Ο συμβολισμός BNF δεν περιλαμβάνει παρενθέσεις. Στο παράδειγμα 5 γράφουμε:

$$\begin{aligned} \langle \text{ακέραιος} \rangle &::= \langle \text{πρόσημο} \rangle \langle \text{ακολουθία} \rangle \\ \langle \text{πρόσημο} \rangle &::= + \mid - \end{aligned}$$

Θα μπορούσαμε εναλλακτικά να γράψουμε:

$$\langle \text{ακέραιος} \rangle ::= + \langle \text{ακολουθία} \rangle \mid - \langle \text{ακολουθία} \rangle$$

Ωστόσο αν γράφαμε:

$$\langle \text{ακέραιος} \rangle ::= (+ \mid -) \langle \text{ακολουθία} \rangle$$

τότε τα σύμβολα (και) θα λαμβάνονταν ως σύμβολα του αλφαβήτου της συντακτικής δομής. Με βάση τον τελευταίο κανόνα (+ και -)34 είναι ακέραιοι!

Παράδειγμα 7. Θα περιγράψουμε σε BNF τη σύνταξη μίας απλής γλώσσας προγραμματισμού που υποστηρίζει έναν τύπο δεδομένων (ακέραιους) και έχει δύο είδη εντολών.

Περιγράφουμε πρώτα τη σύνταξη άτυπα (σε φυσική γλώσσα):

- Το πρόγραμμα αποτελείται από μία ακολουθία εντολών.
- Πριν από κάθε εντολή μπορεί να υπάρχει μία προαιρετική ετικέτα που διαχωρίζεται από την εντολή με το χαρακτήρα `:`.
- Κάθε εντολή τελειώνει με το χαρακτήρα `;`.
- Υπάρχουν δύο είδη εντολών: εντολή ανάθεσης και εντολή άλματος υπό συνθήκη.
- Η εντολή ανάθεσης χρησιμοποιεί τον τελεστή ανάθεσης `:=`. Αριστερά του τελεστή ανάθεσης υπάρχει μία μεταβλητή και δεξιά του μία αριθμητική παράσταση.
- Η εντολή άλματος υπό συνθήκη ξεκινάει με τη λέξη `IF`, ακολουθεί μία αριθμητική παράσταση, η λέξη `GOTO` και μία ετικέτα.

Περιγραφή σύνταξης με BNF

- Μια αριθμητική παράσταση συντάσσεται όπως στα μαθηματικά, από προσημασμένους ακεραίους, μεταβλητες, τους δυαδικούς αριθμητικούς τελεστές $+$, $-$, $*$, $/$ και παρενθέσεις.
- Μια μεταβλητή ξεκινάει με κάποιο από τα γράμματα A,B,C,D,I,J,K,M,N,X,Y,Z και ακολουθεί μία προαιρετική ακολουθία ψηφίων.
- Ένας προσημασμένος ακέραιος έχει τη σύνταξη που περιγράφεται στο παράδειγμα 6.
- Μια ετικέτα ξεκινάει με το γράμμα L και ακολουθεί ένας θετικός αριθμός.
- Η πρώτη εντολή του προγράμματος είναι εντολή ανάθεσης όπου δεξιά του $:=$ υπάρχει η λέξη input.
- Η τελευταία εντολή του προγράμματος είναι εντολή ανάθεσης όπου αριστερά του $:=$ υπάρχει η λέξη output.

Παράδειγμα προγράμματος:

```
N := input;  
X := 1;  
IF N GOTO L1;  
IF 1 GOTO L2;  
L1: X := X * N;  
N := N-1;  
IF N GOTO L1;  
L2: output := X;
```

Περιγραφή σύνταξης με BNF

Σύνταξη της γλώσσας σε BNF:

```
<πρόγραμμα> ::= <πρώτη εντολή> <σώμα> <τελευταία εντολή> |  
                <κεφαλή> output := input ;  
<σώμα> ::= ε | <κεφαλή> <εντολή> ; <σώμα>  
<κεφαλή> ::= ε | <ετικέτα> :  
<εντολή> ::= <εντολή ανάθεσης> | <εντολή άλματος>  
<εντολή ανάθεσης> ::= <μεταβλητή> := <παράσταση>  
<εντολή άλματος> ::= IF <παράσταση> GOTO <ετικέτα>  
<πρώτη εντολή> ::= <κεφαλή> <μεταβλητή> := input ;  
<τελευταία εντολή> ::= <κεφαλή> output := <παράσταση> ;
```

Περιγραφή σύνταξης με BNF

$\langle \text{παράσταση} \rangle ::= \langle \text{ακέραιος} \rangle \mid \langle \text{μεταβλητή} \rangle \mid$
 $\langle \text{παράσταση} \rangle \langle \text{τελεστής} \rangle \langle \text{παράσταση} \rangle \mid$
 $(\langle \text{παράσταση} \rangle)$

$\langle \text{τελεστής} \rangle ::= + \mid - \mid * \mid /$

$\langle \text{μεταβλητή} \rangle ::= \langle \text{γράμμα} \rangle \langle \text{ακολουθία ψηφίων} \rangle$

$\langle \text{γράμμα} \rangle ::= A \mid B \mid C \mid D \mid I \mid J \mid K \mid M \mid N \mid X \mid Y \mid Z$

$\langle \text{ετικέτα} \rangle ::= L \langle \text{θετικός} \rangle$

$\langle \text{ακέραιος} \rangle ::= 0 \mid \langle \text{πρόσημο} \rangle \langle \text{θετικός} \rangle$

$\langle \text{πρόσημο} \rangle ::= + \mid - \mid \varepsilon$

$\langle \text{θετικός} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \langle \text{ακολουθία ψηφίων} \rangle$

$\langle \text{ακολουθία ψηφίων} \rangle ::= \varepsilon \mid \langle \text{ψηφίο} \rangle \langle \text{ακολουθία ψηφίων} \rangle$

$\langle \text{ψηφίο-όχι-0} \rangle ::= 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$\langle \text{ψηφίο} \rangle ::= \langle \text{ψηφίο-όχι-0} \rangle \mid 0$

Περιγραφή σύνταξης με BNF

Οι γλώσσες των οποίων η σύνταξη περιγράφεται σε BNF ονομάζονται γλώσσες χωρίς συμφραζόμενα.

Οι γλώσσες προγραμματισμού συνήθως δεν είναι γλώσσες χωρίς συμφραζόμενα.

Περιγραφή σύνταξης με BNF

Για παράδειγμα δεν μπορούμε να περιγράψουμε σε BNF ότι

- μία μεταβλητή πρέπει να έχει δηλωθεί πριν χρησιμοποιηθεί
- μία μεταβλητή δεν μπορεί να δηλωθεί στο ίδιο μπλοκ περισσότερες από μία φορές
- σε μία ανάθεση οι τύποι πρέπει να είναι συμβατοί

Ωστόσο ο συμβολισμός BNF χρησιμοποιείται για την περιγραφή των συντακτικών χαρακτηριστικών της γλώσσας που είναι ανεξάρτητο από συμφοραζόμενα.

- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα**
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Περιγραφή σύνταξης με συντακτικά διαγράμματα

Τα συντακτικά διαγράμματα προσφέρουν έναν εναλλακτικό, γραφικό τρόπο απεικόνισης της σύνταξης των γλωσσών χωρίς συμφραζόμενα.

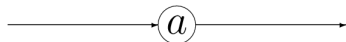
Τα συντακτικά διαγράμματα ορίζονται επαγωγικά.

Περιγραφή σύνταξης με συντακτικά διαγράμματα

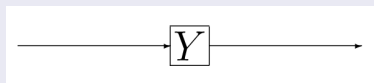
Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει μόνο την κενή συμβολοσειρά.



Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει μόνο την ακολουθία τερματικών συμβόλων a (χρησιμοποιείται κύκλος ή έλλειψη).

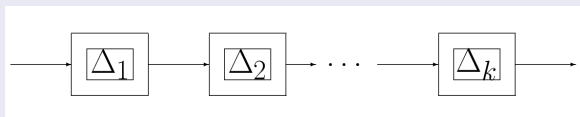


Το παρακάτω συντακτικό διάγραμμα περιγράφει την συντακτική δομή που περιγράφεται και από το συντακτικό διαγραμμα με επικεφαλίδα Y (χρησιμοποιείται ορθογώνιο).



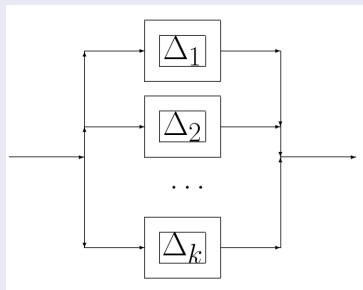
Περιγραφή σύνταξης με συντακτικά διαγράμματα

Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει όλες τις συμβολοσειρές που προκύπτουν από την παράθεση k συμβολοσειρών, οι οποίες παράγονται αντίστοιχα από τα συντακτικά διαγράμματα $\Delta_1, \dots, \Delta_k$.



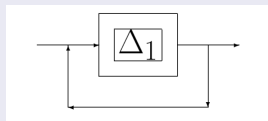
Περιγραφή σύνταξης με συντακτικά διαγράμματα

Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει όλες τις συμβολοσειρές, οι οποίες παράγονται από κάποιο από τα συντακτικά διαγράμματα $\Delta_1, \dots, \Delta_k$.



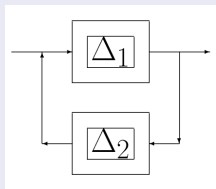
Περιγραφή σύνταξης με συντακτικά διαγράμματα

Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει όλες τις συμβολοσειρές που προκύπτουν από την παράθεση ενός ακεραίου πλήθους συμβολοσειρών, οι οποίες όλες παράγονται από το Δ_1 .



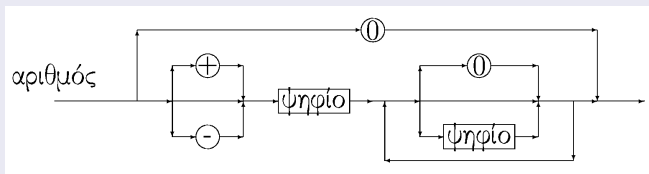
Περιγραφή σύνταξης με συντακτικά διαγράμματα

Το παρακάτω συντακτικό διάγραμμα περιγράφει τη συντακτική δομή που περιλαμβάνει όλες τις συμβολοσειρές που προκύπτουν από την παράθεση ενός περιττού πλήθους συμβολοσειρών, οι οποίες εναλλάξ παράγονται από το Δ_1 και Δ_2 .

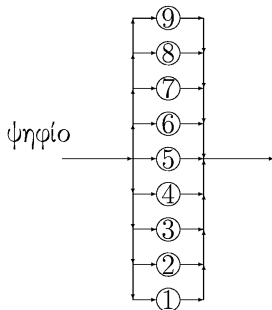


Περιγραφή σύνταξης με συντακτικά διαγράμματα

Παράδειγμα 8. Το παρακάτω συντακτικό διάγραμμα περιγράφει τη σύνταξη των αριθμών, όπως αυτή καθορίζεται στο παράδειγμα 6.



Περιγραφή σύνταξης με συντακτικά διαγράμματα



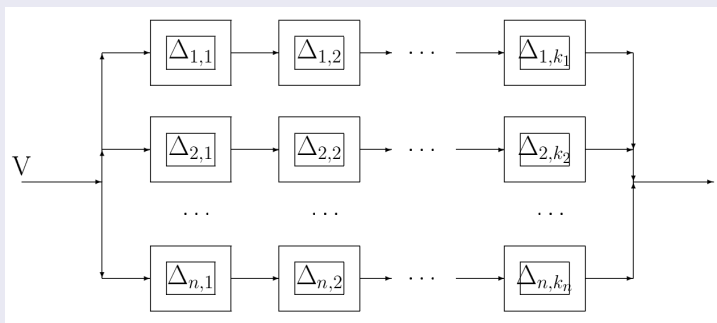
- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα**
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Μετατροπή BNF σε συντακτικά διαγράμματα

Μπορούμε εύκολα να μετατρέψουμε την περιγραφή με συμβολισμό BNF σε συντακτικά διαγράμματα. Για κάθε κανόνα:

$$V ::= X_{1,1} \dots X_{1,k_1} \mid \dots \mid X_{n,1} \dots X_{n,k_n}$$

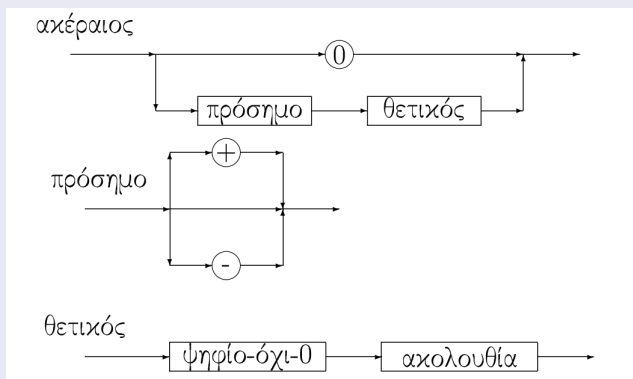
σχεδιάζουμε ένα αντίστοιχο συντακτικό διάγραμμα:



Το $\Delta_{i,j}$ είναι ένα στοιχειώδες συντακτικό διάγραμμα αποτελούμενο από το $X_{i,j}$ μέσα σε κύκλο ή σε ορθογώνιο (ανάλογα με το αν το $X_{i,j}$ είναι τερματικό ή μη τερματικό σύμβολο)

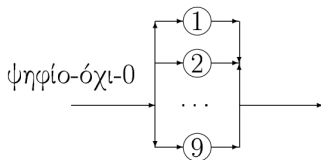
Παράδειγμα

Παράδειγμα 9. Τα παρακάτω συντακτικά διαγράμματα αντιστοιχούν στους κανόνες BNF του παραδείγματος 6.

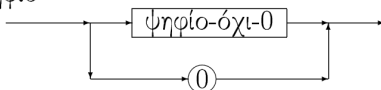


Μετατροπή BNF σε συντακτικά διαγράμματα

ακολουθία



ψηφίο



- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF**
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Στη συνέχεια δείχνουμε πώς μπορούμε να μετατρέψουμε ένα συντακτικό διάγραμμα σε ισοδύναμους κανόνες BNF.

Εξετάζουμε διάφορες περιπτώσεις ανάλογα με τη δομή του συντακτικού διαγράμματος.

Σε κάθε περίπτωση συμβολίζουμε με Y_i τη συντακτική δομή που περιγράφεται από το συντακτικό διάγραμμα Δ_i .

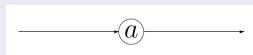
Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από τον κανόνα:

$$\langle X \rangle ::= \epsilon$$

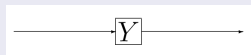
Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από τον κανόνα:

$$\langle X \rangle ::= a$$

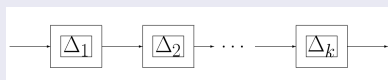
Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από τον κανόνα:

$$\langle X \rangle ::= \langle Y \rangle$$

Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από

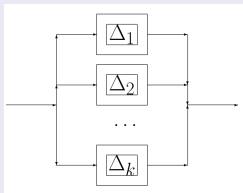
- τον κανόνα

$$\langle X \rangle ::= \langle Y_1 \rangle \langle Y_2 \rangle \dots \langle Y_k \rangle$$

- τους ισοδύναμους κανόνες για τα συντακτικά διαγράμματα $\Delta_1, \Delta_2, \dots, \Delta_k$, οι οποίοι προκύπτουν αναδρομικά.

Μετατροπή συντακτικών διαγραμμάτων σε BNF

Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



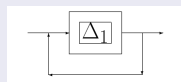
τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από

- τον κανόνα

$$\langle X \rangle ::= \langle Y_1 \rangle \mid \langle Y_2 \rangle \mid \dots \mid \langle Y_k \rangle$$

- τους ισοδύναμους κανόνες για τα συντακτικά διαγράμματα $\Delta_1, \Delta_2, \dots, \Delta_k$, οι οποίοι προκύπτουν αναδρομικά.

Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



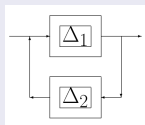
τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από

- τον κανόνα

$$\langle X \rangle ::= \langle Y_1 \rangle \mid \langle Y_1 \rangle \langle X \rangle$$

- τους ισοδύναμους κανόνες για το συντακτικό διάγραμμα Δ_1 , οι οποίοι προκύπτουν αναδρομικά.

Αν η συντακτική δομή X περιγράφεται από ένα συντακτικό διάγραμμα της μορφής:



τότε η ισοδύναμη περιγραφή σε BNF αποτελείται από

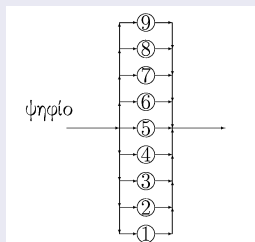
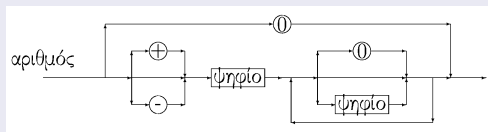
- τον κανόνα

$$\langle X \rangle ::= \langle Y_1 \rangle \mid \langle Y_1 \rangle \langle Y_2 \rangle \langle X \rangle$$

- τους ισοδύναμους κανόνες για τα συντακτικά διαγράμματα Δ_1 και Δ_2 , οι οποίοι προκύπτουν αναδρομικά.

Παράδειγμα

Παράδειγμα 10. Η παρακάτω περιγραφή αριθμών σε BNF προκύπτει από το συντακτικό διάγραμμα του παραδείγματος 8:



Μετατροπή συντακτικών διαγραμμάτων σε BNF

```
<αριθμός> ::= <A1> | <A2>  
<A1> ::= 0  
<A2> ::= <B1> <B2> <B3>  
<B1> ::= <C1> | <C2> | <C3>  
<C1> ::= +  
<C2> ::= ε  
<C3> ::= -  
<B2> ::= <ψηφίο>  
<B3> ::= <D1> | <D1> <B3>
```



```
<D1> ::= <E1> | <E2> | <E3>  
<E1> ::= 0  
<E2> ::= ε  
<E3> ::= <ψηφίο>  
<ψηφίο> ::= <F1> | <F2> | ... | <F9>  
<F1> ::= 1  
<F2> ::= 2  
...  
<F9> ::= 9
```

Μία προφανής βελτίωση ώστε να μειωθεί το πλήθος των κανόνων είναι η απαλοιφή των κανόνων που προκύπτουν από απλά συντακτικά διαγράμματα.

Αυτό μπορεί να γίνει με απ' ευθείας αντικατάσταση των μη τερματικών συμβόλων που αντιστοιχούν σε αυτά τα διαγράμματα στα σώματα των κανόνων όπου εμφανίζονται.

Μετατροπή συντακτικών διαγραμμάτων σε BNF

```
<αριθμός> ::= 0 | <A2>  
<A2> ::= <B1> <ψηφίο> <B3>  
<B1> ::= + | ε | -  
<B3> ::= <D1> | <D1> <B3>  
<D1> ::= 0 | ε | <ψηφίο>  
<ψηφίο> ::= 1 | 2 | ... | 9
```

- 1 Σύμβολα, Αλφάβητα και Συντακτικές Δομές
- 2 Περιγραφή σύνταξης με BNF (Backus-Naur Form)
- 3 Περιγραφή σύνταξης με συντακτικά διαγράμματα
- 4 Μετατροπή BNF σε συντακτικά διαγράμματα
- 5 Μετατροπή συντακτικών διαγραμμάτων σε BNF
- 6 Σημασιολογία Γλωσσών Προγραμματισμού

Ο καθορισμός της σημασιολογίας μιας γλώσσας προγραμματισμού είναι πολύ πιο δύσκολος από τον καθορισμό της σύνταξης.

Για αυτό το λόγο σε πολλές περιπτώσεις, χρησιμοποιείται η φυσική γλώσσα για τη περιγραφή της σημασιολογίας.

Ωστόσο έχουν προταθεί διάφοροι εναλλακτικοί τυπικοί τρόποι περιγραφής της σημασιολογίας γλωσσών προγραμματισμού, οι οποίοι συνοψίζονται παρακάτω.

Λειτουργική ή ερμηνευτική σημασιολογία (operational ή interpretive semantics):

- Η σημασία ενός προγράμματος δίνεται περιγράφοντας την εκτέλεση του σε μία ιδεατή μηχανή.
- Οι εντολές της γλώσσας προγραμματισμού ερμηνεύονται ως μία σειρά εντολών της ιδεατής μηχανής.

Ενδεικτική σημασιολογία (denotational semantics):

- Είναι η πιο αφηρημένη μέθοδος περιγραφής.
- Ορίζει ένα σύνολο από μαθηματικά αντικείμενα για κάθε συντακτική δομή της γλώσσας, καθώς και μία συνάρτηση που απεικονίζει τα στοιχεία της συντακτικής δομής σε μαθηματικά αντικείμενα.

Αξιωματική σημασιολογία (axiomatic semantics):

- Στηρίζεται στη μαθηματική λογική και αναπτύχθηκε αρχικά με σκοπό την τυπική επαλήθευση προγραμμάτων.
- Αντιστοιχεί σε κάθε εντολή του προγράμματος ένα αξίωμα ή έναν κανόνα εξαγωγής συμπεράσματος, ο οποίος καθορίζει τις συνθήκες που ισχύουν μετά την εκτέλεση της εντολής, με βάση τις συνθήκες που ισχύουν πριν την εκτέλεση της εντολής

Σημασιολογία ελάχιστου μοντέλου (minimum model semantics):

- Χρησιμοποιείται στις γλώσσες λογικού προγραμματισμού.
- Η σημασία του προγράμματος περιγράφεται με το σύνολο των ατομικών προτάσεων που θεωρούνται αληθείς με βάση το πρόγραμμα.

Σημασιολογία σταθερού σημείου (fixpoint semantics):

- Χρησιμοποιείται στις γλώσσες λογικού προγραμματισμού.
- Σε κάθε πρόγραμμα αντιστοιχεί ένας τελεστής T_P ο οποίος απεικονίζει σύνολα ατομικών προτάσεων σε σύνολα ατομικών προτάσεων. Η σημασία του προγράμματος ορίζεται ως το ελάχιστο σύνολο των προτάσεων M για το οποίο $T_P(M) = M$.