

Σύνθετη Συνθήκη

Οι τελεστές ΚΑΙ & Ή χρησιμοποιούνται όταν χρειάζονται να συνδυαστούν περισσότερες από μια λογικές συνθήκες, ενώ ο τελεστής ΟΧΙ εφαρμόζεται σε μία λογική συνθήκη.

Σύνθετη συνθήκη εννοούμε μία συνθήκη που αποτελείται από απλές συνθήκες που συνδέονται με τους τελεστές ΚΑΙ & Ή. Τέτοιες συνθήκες είναι για παράδειγμα: $\alpha > \beta$ ΚΑΙ $\gamma > \delta$, $\alpha \leq \varepsilon$ Ή $\beta < \gamma$, κ.λπ..

Ο τελεστής ΟΧΙ σημαίνει ότι η συνθήκη που ελέγχεται είναι η συμπληρωματική της. Για παράδειγμα η συνθήκη ΟΧΙ $\alpha > \beta$ γράφεται σαν $\alpha \leq \beta$.

Οι παρακάτω πίνακες μας δείχνουν πότε μία σύνθετη συνθήκη είναι αληθής (ισχύει) ή ψευδής (δεν ισχύει).

Τελεστής ΚΑΙ		
Συνθήκη 1	Συνθήκη 2	Συνθήκη 1 ΚΑΙ Συνθήκη 2
A	A	A
A	Ψ	Ψ
Ψ	A	Ψ
Ψ	Ψ	Ψ

Τελεστής Ή		
Συνθήκη 1	Συνθήκη 2	Συνθήκη 1 Ή Συνθήκη 2
A	A	A
A	Ψ	A
Ψ	A	A
Ψ	Ψ	Ψ

Τελεστής ΟΧΙ	
Συνθήκη 1	ΟΧΙ Συνθήκη 1
A	Ψ
Ψ	A

Στα αρχικά στάδια σχεδιασμού αλγορίθμων θα πρέπει να αποφεύγεται η σύνταξη σύνθετων συνθηκών. Άλλωστε κάθε σύνθετο Αν μπορεί να αναλυθεί σε απλούστερα περιεχόμενα Αν.

Φίλτρο τιμών

Στην περίπτωση εισαγωγής στοιχείων από το πληκτρολόγιο υπάρχει ανάγκη ελέγχου των δεδομένων, πριν αυτά εισαχθούν στον αλγόριθμο, αν πληρούν τους περιορισμούς της άσκησης.

Γενικά υπάρχουν δύο τρόποι αντιμετώπισης:

- Μια επιπλέον συνθήκη μέσα στην διαδικασία ώστε να αναγνωριστεί το λανθασμένο δεδομένο, και να μην συμμετάσχει σε διαδικασία που δεν προβλέπει τέτοια τιμή. Για παράδειγμα διαίρεση με μηδέν.
- Μια επιπλέον διαδικασία στην είσοδο των δεδομένων ώστε να διαπιστωθούν τα λάθη τιμών, να απορριφθούν και να ζητηθούν νέα, έγκυρα.

Ο δεύτερος τρόπος είναι ο ενδεδειγμένος δεδομένου ότι διαχωρίζει την επεξεργασία από τον έλεγχο, προσεγγίζοντας στις αρχές του δομημένου προγραμματισμού. Αυτός αφορά την δημιουργία φίλτρου τιμών

Αυτός ο έλεγχος θα πρέπει να λειτουργεί με τρόπο απαγορευτικό για τον χρήστη, να μην του επιτρέπει δηλαδή να εισάγει δεδομένα που είναι εκτός προδιαγραφών, και άρα δεν θα μπορέσει να τα επεξεργαστεί ο αλγόριθμος.

Επομένως ο έλεγχος θα γίνεται με μία από τις δύο γνωστές επαναληπτικές δομές. Η συνθήκη της δομής θέλει προσεκτική σύνταξη που θα γίνει εύκολα αν χρησιμοποιηθεί ο πίνακας αληθείας των συνθηκών που θα ελέγχονται.

Ένα φίλτρο δημιουργείται από τις δύο γνωστές επαναληπτικές δομές. Η συνθήκη που θα αποκλείει τις λανθασμένες τιμές θέλει προσεκτική σύνταξη δεδομένου ότι τις περισσότερες φορές θα είναι σύνθετη. Σε αυτή την περίπτωση είναι σκόπιμο να χρησιμοποιηθεί ο πίνακας αληθείας των συνθηκών που θα ελέγχονται.

Για παράδειγμα, πρέπει να ελεγχθεί αν ένας αριθμός είναι φυσικός (ακέραιος, θετικός, $\neq 0$). Ο παρακάτω πίνακας θα βοηθήσει στην σωστή σύνταξη της συνθήκης στην επαναληπτική δομή.

$x \neq \text{int}(x)$	$x \leq 0$	KAI	Ή
A	A	A	A
A	Ψ	Ψ	A
Ψ	A	Ψ	A
Ψ	Ψ	Ψ	Ψ

$x = \text{int}(x)$	$x > 0$	KAI	Ή
A	A	A	A
A	Ψ	Ψ	A
Ψ	A	Ψ	A
Ψ	Ψ	Ψ	Ψ

Άρα η συνθήκη για την οποία θα τερματίζει μία επαναληπτική διαδικασία τύπου "Όσο" είναι η διάζευξη των δύο συνθηκών ($x \neq \text{int}(x)$ Ή $x \leq 0$). Ενώ αν επιλεγεί η δομή "Μέχρι" η συνθήκη θα είναι ($x = \text{int}(x)$ KAI $x > 0$).

Η σύνταξη ενός πίνακα αποφάσεων δεν είναι πάντα εύκολη υπόθεση. Για παράδειγμα ο προηγούμενος πίνακας θα μπορούσε να είναι να έχει άλλους συνδυασμούς των δύο συνθηκών που δεν οδηγούν πουθενά.

Η επαναληπτική διαδικασία.

Η επαναληπτική διαδικασία είναι η ισχυρότερη δομή σε ένα αλγόριθμο. Μία επαναληπτική διαδικασία μπορεί να γίνεται με δύο τρόπους:

- Με έλεγχο επί των δεδομένων.
- Με έλεγχο βοηθητικής μεταβλητής
 - ο Με ερώτηση – επανάληψη
 - ο Με ορισμένο αριθμό επαναλήψεων.

Σε κάθε περίπτωση ο έλεγχος της επανάληψης πρέπει να γράφεται μαζί με την αλλαγή της τιμής του πεδίου στο οποίο στηρίζεται η επανάληψη.

Ειδικά στην περίπτωση του ελέγχου επί των δεδομένων δεν πρέπει να μεσολαβεί καμία εντολή μεταξύ της εισόδου του δεδομένου και του ελέγχου της τιμής.

Υπάρχουν δύο επαναληπτικές δομές:

Όσο συνθήκη1 επανάλαβε εντολές Τέλος επανάληψης	Αρχή επανάληψης εντολές Μέχρι συνθήκη2 επανάλαβε
---	--

Οι δύο δομές μπορούν να χρησιμοποιηθούν η μία αντί της άλλης χωρίς κανένα τεχνικό περιορισμό. Όταν συμβαίνει αυτό η συνθήκη1 έχει συμπληρωματική της την συνθήκη 2.

Προσοχή η δομή:

Για μεταβλητή από μεταβλητή1/σταθερά1 **μέχρι** μεταβλητή2/σταθερά2 [**με βήμα** σταθερά3] **επανάλαβε**

εντολές

Τέλος επανάληψης

Δεν αποτελεί ξεχωριστή επαναληπτική δομή αλλά είναι ίδια με την δομή ΟΣΟ, μόνο που περιέχει αυτόματη αύξηση της μεταβλητής.

Δηλαδή:

$i \leftarrow \text{σταθερά1}$ Όσο $i \leq \text{σταθερά2}$ επανάλαβε $i \leftarrow i + \text{σταθερά3}$ Τέλος επανάληψης	Για i από σταθερά1 μέχρι σταθερά2 με βήμα σταθερά3 επανάλαβε Τέλος επανάληψης
---	---

Γενικά η Για χρησιμοποιείται όταν είναι γνωστός ο αριθμός επαναλήψεων.

Η συνθήκη που ελέγχεται σε μία επαναληπτική δομή πρέπει να αναφέρεται σε μεταβλητή / μεταβλητές που:

- Έχει πάρει ήδη μία τιμή. (αλλιώς ο έλεγχος είναι πρακτικά αδύνατος, γιατί ο αλγόριθμος δεν μπορεί να διαπιστώσει αν η συνθήκη ισχύει ή όχι).
- Αλλάζει τιμή μέσα στην επαναληπτική διαδικασία. (αλλιώς η διαδικασία δεν πρόκειται να τερματιστεί ποτέ αφού η συνθήκη θα παραμένει αληθής ή ψευδής, ανάλογα με την περίπτωση, που θα είναι η ίδια με αυτήν για την οποία ξεκίνησε η διαδικασία).

Σημείωση:

- Η δομή ΟΣΟ επαναλαμβάνεται όταν η συνθήκη είναι ΑΛΗΘΗΣ. Μπορεί να μην εκτελεστεί ΠΟΤΕ.
- Η δομή ΜΕΧΡΙ επαναλαμβάνεται όταν η συνθήκη είναι ΨΕΥΔΗΣ. Θα εκτελεστεί ΟΠΩΣ-ΔΗΠΟΤΕ μία φορά.
- Στην δομή ΓΙΑ δεν επιτρέπεται να μεταβληθεί η τιμή της μεταβλητής. Θα εκτελεστεί τόσες φορές, όσες ακριβώς έχει οριστεί.
- Κάθε διαδικασία Όσο μπορεί να μετατραπεί σε διαδικασία Μέχρι, αν η συνθήκη που ελέγχει την επανάληψη της Όσο, μετατραπεί στην συμπληρωματική της.

Άσκηση: Με ποιο τρόπο μια διαδικασία Όσο μετατρέπεται σε διαδικασία Για. Και πως γίνεται το αντίστροφο. Γενικά κάτω από ποιες συνθήκες μπορούμε να μετατρέπουμε μια επαναληπτική διαδικασία σε μία άλλη.