

Μάθημα 34

Δομή επανάληψης ΜΕΧΡΙΣ_ΟΤΟΥ

Πρόβλημα με διπλό υπόλοιπο / ΣΑΒΒΑΣ

(ΣΑΒΒΑΣ) Το φορτηγάκι μίας μεταφορικής εταιρείας, μπορεί να μεταφέρει το πολύ 100 κουτιά τυποποιημένου μεγέθους, αρκεί το συνολικό τους βάρος να μην ξεπερνά τα 800 κιλά.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο:

Για κάθε παραγγελία που πρόκειται να φορτωθεί, να διαβάσει το όνομα του πελάτη, το πλήθος των κουτιών της, καθώς και το συνολικό τους βάρος σε κιλά (δεν απαιτείται έλεγχος εγκυρότητας) και στην περίπτωση που και οι δύο τιμές (πλήθους και βάρους) είναι εντός ορίων, να "φορτώνει" την παραγγελία.

Να τερματίζει τη διαδικασία φόρτωσης των παραγγελιών όταν διαβαστεί είτε πλήθος κουτιών, είτε βάρος κουτιών με τα οποία θα ξεπεραστούν τα όρια που αναφέρθηκαν για το φορτηγάκι.

Μετά τη διαδικασία φόρτωσης των παραγγελιών, να εμφανίζει:

- α) Το μέσο πλήθος κουτιών των παραγγελιών που φορτώθηκαν.
- β) Το όνομα του πελάτη που έκανε την παραγγελία με το μικρότερο βάρος (έστω ότι είναι μοναδική).

Να θεωρήσετε ότι θα φορτωθεί τουλάχιστον μία παραγγελία.

ΠΡΟΓΡΑΜΜΑ μάθημα_34_ΔΥΟ_ΥΠΟΛΟΙΠΑ
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: C, S, Y1, ΚΟΥΤΙΑ

ΠΡΑΓΜΑΤΙΚΕΣ: Y2, ΒΑΡΟΣ, min

ΧΑΡΑΚΤΗΡΕΣ: ON, pos

ΑΡΧΗ

C <- 0

min <- 801

Y1 <- 100 ! Κουτιά

Y2 <- 800 ! Κιλά

ΔΙΑΒΑΣΕ ON, ΚΟΥΤΙΑ, ΒΑΡΟΣ

ΟΣΟ ΚΟΥΤΙΑ <= Y1 **ΚΑΙ** ΒΑΡΟΣ <= Y2 **ΕΠΑΝΑΛΑΒΕ**

C <- C + 1

Y1 <- Y1 - ΚΟΥΤΙΑ

Y2 <- Y2 - ΒΑΡΟΣ

ΑΝ ΒΑΡΟΣ < min **ΤΟΤΕ**

min <- ΒΑΡΟΣ

pos <- ON

ΤΕΛΟΣ_ΑΝ

ΔΙΑΒΑΣΕ ON, ΚΟΥΤΙΑ, ΒΑΡΟΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

S <- 100 - Y1

ΓΡΑΨΕ S / C, pos

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Επαναληπτικό σχήμα με έλεγχο στο τέλος (ΜΕΧΡΙΣ_ΟΤΟΥ)

Χρησιμοποιείται όταν θέλουμε να εκτελέσουμε επαναληπτικά κάποιες εντολές, μέχρις ότου να γίνει αληθινή μία συνθήκη (η οποία βρίσκεται στο τέλος της επανάληψης).

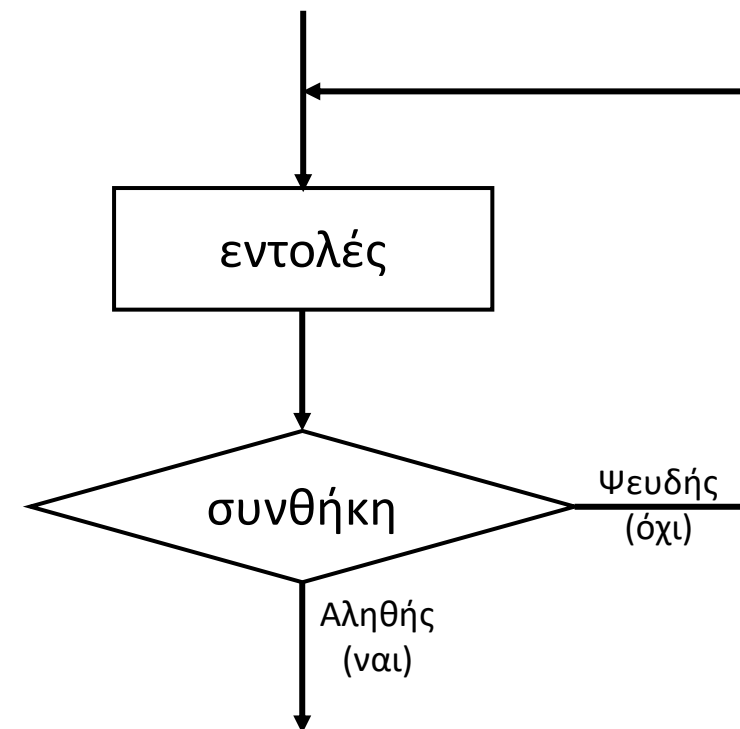
Συντάσσεται ως εξής:

Αρχή_επανάληψης

<εντολές>

Μέχρις_ότου <συνθήκη>

Το αντίστοιχο διάγραμμα ροής είναι:



Λειτουργία: Εκτελούνται πρώτα οι εντολές του βρόγχου.

Έπειτα ελέγχεται η συνθήκη.

Αν η συνθήκη είναι ΑΛΗΘΗΣ τότε τερματίζεται ο βρόγχος και εκτελείται η εντολή που βρίσκεται μετά το Μέχρις_ότου.

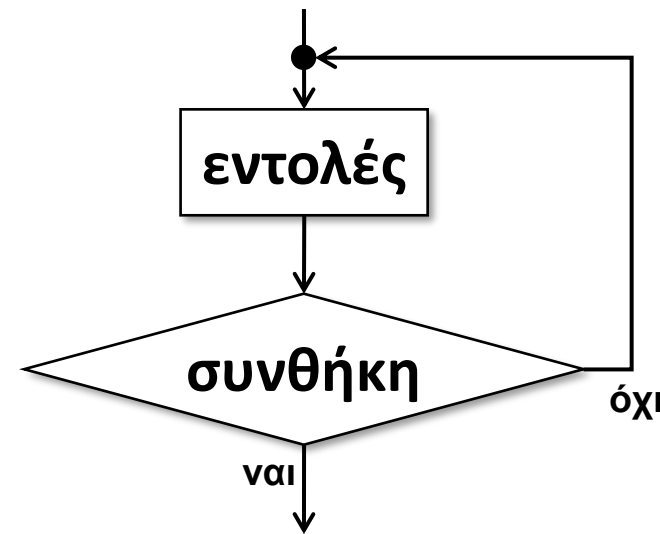
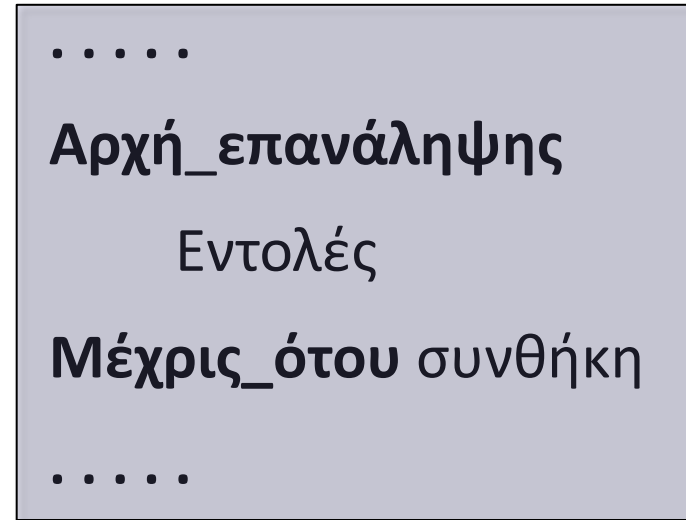
Αν όμως η συνθήκη είναι ΨΕΥΔΗΣ τότε εκτελούνται ξανά οι εντολές του βρόγχου.

Προσοχή! Οι εντολές του βρόγχου θα εκτελεστούν τουλάχιστον μία φορά.

Επαναληπτικό σχήμα με έλεγχο στο τέλος (ΜΕΧΡΙΣ_ΟΤΟΥ) / Σχήμα και λειτουργία

Χρησιμοποιείται όταν θέλουμε να εκτελέσουμε επαναληπτικά κάποιες εντολές μέχρι να γίνει αληθινή μία συνθήκη (η οποία βρίσκεται στο τέλος της επανάληψης).

- Σχήμα και διάγραμμα ροής



Εκτελούνται οι εντολές.

Ελέγχεται η συνθήκη.

Εάν η συνθήκη δεν ισχύει (είναι ψευδής) εκτελούνται πάλι οι εντολές.

Εάν η συνθήκη ισχύει (είναι αληθής) βγαίνουμε από τη δομή.

Επαναληπτικό σχήμα με έλεγχο στο τέλος (ΜΕΧΡΙΣ_ΟΤΟΥ) / Παράδειγμα

Αλγόριθμος που διαβάζει και εκτυπώνει μη μηδενικούς αριθμούς.

← 3

← -1

← 0

Αλγόριθμος δομή_μέχρι_1

Αρχή_επανάληψης

Διάβασε X

Εκτύπωσε X

Μέχρις_ότου (X = 0) ❌

Τέλος δομή_μέχρι_1

επανάληψη	X	εκτύπωση
1	3	
2		3

Η μεταβλητή που ελέγχεται στη συνθήκη της δομής επανάληψης Μέχρις_ότου (εδώ είναι η X), πρέπει να τροποποιείται μέσα στο βρόχο (εδώ μέσω του **Διάβασε X**).

Η δομή Μέχρις_ότου θα εκτυπώσει και το 0.

Δομή επανάληψης ΜΕΧΡΙΣ_ΟΤΟΥ / παράδειγμα 1

Να γίνουν ο πίνακας τιμών και το διάγραμμα ροής, για το παρακάτω πρόγραμμα, όταν στο X δοθούν διαδοχικά οι τιμές: 10, 15, 20.

ΠΡΟΓΡΑΜΜΑ μάθημα_34_ΠΑΡΑΔΕΙΓΜΑ_1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: X, C, S

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΑΡΧΗ

C ← 0

S ← 0

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ X

C ← C + 1

S ← S + X

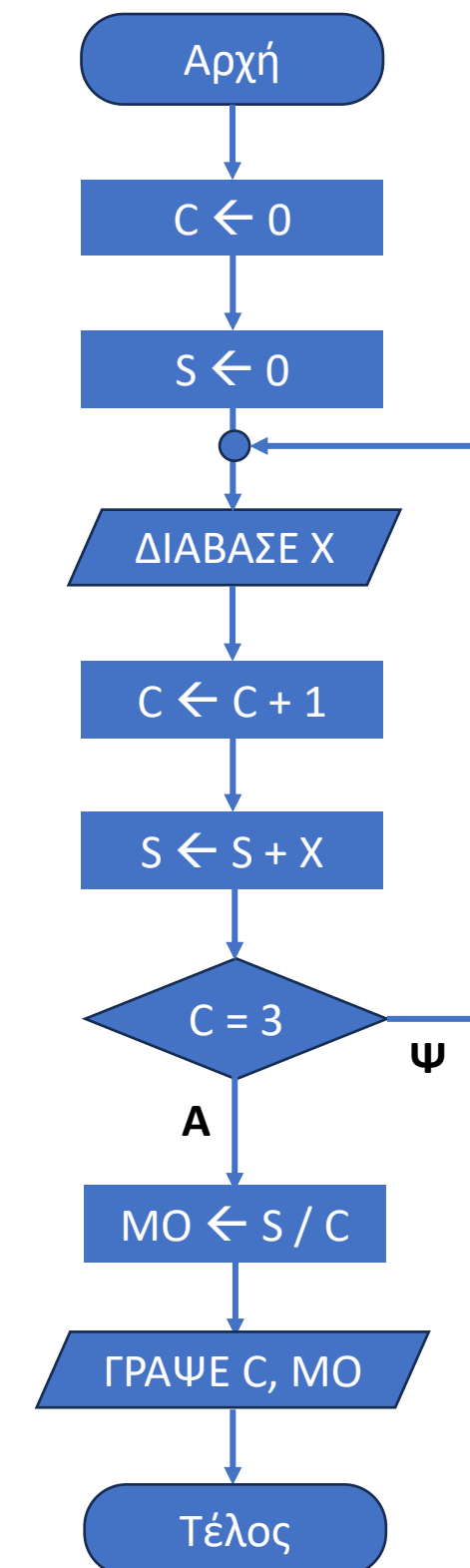
ΜΕΧΡΙΣ_ΟΤΟΥ C = 3

ΜΟ ← S / C

ΓΡΑΨΕ C, ΜΟ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

X	C	S	C = 3	ΜΟ	Έξοδος
	0	0			
10	1	10	ΨΕΥΔΗΣ		
15	2	25	ΨΕΥΔΗΣ		
20	3	45	ΑΛΗΘΗΣ		
				15.00	3, 15.00



Δομή επανάληψης ΜΕΧΡΙΣ_ΟΤΟΥ / παράδειγμα 2

Το παρακάτω πρόγραμμα επιχειρεί να διαβάσει αριθμούς, μέχρι να δοθεί ως τιμή το μηδέν και να υπολογίζει και εμφανίζει το πλήθος και το μέσο όρο των μη μηδενικών αριθμών που δόθηκαν, θεωρώντας πως θα δοθεί ένας τουλάχιστον μη μηδενικός αριθμός.

Μέσα από έναν πίνακα τιμών και δίνοντας διαδοχικά τις τιμές 13, 7, 10, 0 να διαπιστώσετε ότι δεν επιτελεί την επιθυμητή λειτουργία και προτείνετε κάποια διόρθωση.

ΠΡΟΓΡΑΜΜΑ μάθημα_34_ΠΑΡΑΔΕΙΓΜΑ_2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: X, C, S

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΑΡΧΗ

C ← 0

S ← 0

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ X

ΑΝ X <> 0 **ΤΟΤΕ**

C ← C + 1

S ← S + X

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ X = 0

ΜΟ ← S / C

ΓΡΑΨΕ C, ΜΟ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

X	C	S	X = 0	ΜΟ	Έξοδος
	0	0			
13	1	13	ΨΕΥΔΗΣ		
7	2	20	ΨΕΥΔΗΣ		
10	3	30	ΨΕΥΔΗΣ		
0	4	30	ΑΛΗΘΗΣ	7.50	4, 7.50

X	X <> 0	C	S	X = 0	ΜΟ	Έξοδος
		0	0			
13	ΑΛΗΘΗΣ	1	13	ΨΕΥΔΗΣ		
7	ΑΛΗΘΗΣ	2	20	ΨΕΥΔΗΣ		
10	ΑΛΗΘΗΣ	3	30	ΨΕΥΔΗΣ		
0	ΨΕΥΔΗΣ			ΑΛΗΘΗΣ	10.00	3, 10.00

Δομή επανάληψης ΜΕΧΡΙΣ_ΟΤΟΥ / παράδειγμα 3

Να αναπτυχθεί αλγόριθμος που θα διαβάζει επαναληπτικά θετικές τιμές και θα εμφανίζει τις τιμές αυτές, ακολουθούμενες από το λογάριθμό τους.

Η επανάληψη θα τερματίζεται όταν δοθεί αριθμός που δεν είναι θετικός, οπότε και εμφανίζεται το μήνυμα «Τερματισμός».

Αλγόριθμος μάθημα_34_ΠΑΡΑΔΕΙΓΜΑ_3Α

Διάβασε X

Όσο $X > 0$ επανάλαβε

Εμφάνισε X, $\text{LOG}(X)$

Διάβασε X

Τέλος_επανάληψης

Εμφάνισε "Τερματισμός"

Τέλος μάθημα_34_3Α

Αλγόριθμος μάθημα_34_ΠΑΡΑΔΕΙΓΜΑ_3Β

Αρχή_επανάληψης

Διάβασε X

Αν $X > 0$ τότε

Εμφάνισε X, $\text{LOG}(X)$

Τέλος_αν

Μέχρις_ότου $X \leq 0$

Εμφάνισε "Τερματισμός"

Τέλος μάθημα_34_3Β

Δομή επανάληψης ΜΕΧΡΙΣ_ΟΤΟΥ / παράδειγμα 4

Να αναπτυχθεί πρόγραμμα που θα διαβάζει επαναληπτικά αριθμούς, μέχρι να δοθεί μία τιμή που δεν είναι θετική και θα εμφανίζει το πλήθος των τιμών (θετικών) που δόθηκαν, και στην περίπτωση που δόθηκε μία τουλάχιστον θετική τιμή, τη μεγαλύτερη από αυτές, καθώς και σε ποια επανάληψη καταχωρήθηκε η τιμή αυτή για τελευταία φορά.

ΠΡΟΓΡΑΜΜΑ μάθημα_34_ΠΑΡΑΔΕΙΓΜΑ_4

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: C, pos

ΠΡΑΓΜΑΤΙΚΕΣ: X, max

ΑΡΧΗ

C <- 0

max <- 0

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ X

ΑΝ X > 0 **ΤΟΤΕ**

C <- C + 1

ΑΝ X >= max **ΤΟΤΕ**

max <- X

pos <- C

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ X <= 0

ΓΡΑΨΕ 'Δόθηκαν ', C, ' αριθμοί'

ΑΝ C <> 0 **ΤΟΤΕ**

ΓΡΑΨΕ 'Μεγαλύτερη τιμή: ', max

ΓΡΑΨΕ 'στην ', pos, ' η επανάληψη'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Ενότητα 3

Ασκήσεις

8 / σελίδα 93

(να γίνει και διάγραμμα ροής)

6 / σελίδα 101