

Ανάπτυξη εφαρμογών σε
προγραμματιστικό περιβάλλον
Γ' Λυκείου

ΚΕΦΑΛΑΙΟ 7

Βασικές έννοιες Προγραμματισμού



Χρήστος Μουρατίδης - Έκδοση 2020

mouratx@yahoo.com

<http://users.sch.gr/mouratx>

Περιεχόμενα

Η ΓΛΩΣΣΑ - ΜΙΑ ΥΠΟΘΕΤΙΚΗ ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ	1
ΤΟ ΑΛΦΑΒΗΤΟ ΤΗΣ ΓΛΩΣΣΑΣ.....	1
ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ	2
ΣΤΑΘΕΡΕΣ	3
ΜΕΤΑΒΛΗΤΕΣ	5
ΑΡΙΘΜΗΤΙΚΟΙ ΤΕΛΕΣΤΕΣ	6
ΣΥΝΑΡΤΗΣΕΙΣ	7
ΑΡΙΘΜΗΤΙΚΕΣ ΕΚΦΡΑΣΕΙΣ.....	8
ΕΝΤΟΛΗ ΕΚΧΩΡΗΣΗΣ	10
ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ-ΕΞΟΔΟΥ	11
ΔΟΜΗ ΠΡΟΓΡΑΜΜΑΤΟΣ	13
ΠΑΡΑΔΕΙΓΜΑΤΑ ΠΡΟΓΡΑΜΜΑΤΩΝ ΣΤΗ ΓΛΩΣΣΑ	14
ΠΑΡΑΔΕΙΓΜΑ 1.....	14
ΠΑΡΑΔΕΙΓΜΑ 2.....	16
ΠΑΡΑΔΕΙΓΜΑ 3.....	17
ΕΡΩΤΗΣΕΙΣ ΚΑΤΑΝΟΗΣΗΣ.....	18

Η ΓΛΩΣΣΑ - Μία υποθετική γλώσσα προγραμματισμού

Για την εκμάθηση των βασικών στοιχείων προγραμματισμού και την ανάπτυξη των προγραμμάτων, θα χρησιμοποιήσουμε μία «φανταστική» γλώσσα προγραμματισμού **υψηλού επιπέδου** που θα την ονομάσουμε **ΓΛΩΣΣΑ**.

Η ΓΛΩΣΣΑ είναι σχεδιασμένη για εκπαιδευτικούς σκοπούς. Περιέχει χαρακτηριστικά (αλφάβητο, λεξιλόγιο), δομές και εντολές που βρίσκονται στις πραγματικές σύγχρονες γλώσσες προγραμματισμού, όπως η Pascal, C, Java, Visual Basic και άλλες, χωρίς να ασχολείται με τις τεχνικές λεπτομέρειες αυτών. Έτσι, ο αναγνώστης επικεντρώνεται στην ανάπτυξη του αλγορίθμου του και την μετατροπή αυτού σε σωστό πρόγραμμα.

Στα επόμενα θα ακολουθήσουμε μερικές **τυπογραφικές συμβάσεις**:

- Οι **εντολές** και γενικά οι **δεσμευμένες λέξεις της ΓΛΩΣΣΑΣ** θα είναι με **μπλε χρώμα** και **πάντα με κεφαλαία**. Π.χ. **ΓΡΑΨΕ**, **ΜΕΤΑΒΛΗΤΕΣ**, **ΑΡΧΗ**
- Οι **μεταβλητές** θα είναι με **πεζά ή κεφαλαία**, αλλά το πρώτο γράμμα του ονόματος της μεταβλητής θα είναι πάντα κεφαλαίο. Π.χ. **Άθροισμα**, **ΑΘΡΟΙΣΜΑ**, **Μέσος_Όρος**, **Μέσος_Όρος**.
- Τα **σχόλια** των προγραμμάτων θα είναι με **πράσινο χρώμα**.

Το αλφάβητο της ΓΛΩΣΣΑΣ

Το αλφάβητο της ΓΛΩΣΣΑΣ περιέχει:

Γράμματα:

Κεφαλαία του ελληνικού αλφαβήτου (Α-Ω)

Πεζά του ελληνικού αλφαβήτου (α-ω)

Κεφαλαία του λατινικού αλφαβήτου (Α-Z)

Πεζά του λατινικού αλφαβήτου (a-z)

Ψηφία:

0-9

Ειδικοί χαρακτήρες:

+ - * / = . () . , ' ! & κενός χαρακτήρας ^

Τύποι δεδομένων

Ένας τύπος δεδομένων προσδιορίζει το είδος των δεδομένων (π.χ. αριθμοί, χαρακτήρες κλπ) που μπορεί να επεξεργαστεί μία γλώσσα προγραμματισμού, εδώ η ΓΛΩΣΣΑ.

Οι τύποι δεδομένων που υποστηρίζει η ΓΛΩΣΣΑ είναι οι εξής:

Τύπος δεδομένων	Περιγραφή	Παράδειγμα
Ακέραιος¹	Περιλαμβάνει τους αρνητικούς, θετικούς ακεραίους και το 0.	-20, 35, 0.
Πραγματικός	Περιλαμβάνει τους αρνητικούς, θετικούς πραγματικούς και το 0.	-20.45, 35.0, 35.62, 3.14, 0.

¹ Τα δεδομένα καταχωρούνται στη μνήμη του υπολογιστή καταλαμβάνοντας έναν ορισμένο αριθμό από bytes. Στις γλώσσες προγραμματισμού κάθε τύπος έχει παραλλαγές για την αποτελεσματικότερη διαχείριση των δεδομένων. Για παράδειγμα, στην γλώσσα Visual Basic, ένα δεδομένο του τύπου Int32 καταλαμβάνει 32 bytes, ένα δεδομένο του τύπου Int64 καταλαμβάνει 64 bytes.

Όσο μεγαλύτερο ο αριθμός των bytes τόσο μεγαλύτερο και το εύρος τιμών που μπορεί να καλύψει ο συγκεκριμένος τύπος δεδομένων.

Χαρακτήρας	Περιλαμβάνει έναν αλφαριθμητικό χαρακτήρα ² ή μία σειρά αλφαριθμητικών χαρακτήρων . Οι χαρακτήρες πρέπει να βρίσκονται μέσα σε απλά εισαγωγικά, '' .	'Χρήστος' 'X' 'AB25' 'AB 25' '12' '12.34' 'Παρακαλώ, δώστε έναν αριθμό' ' '(ένας χαρακτήρας space), '' (υποδηλώνει τον κενό χαρακτήρα). Μία χαρακτηριστική περίπτωση είναι τα πληροφοριακά μηνύματα που εξάγει ο υπολογιστής.
Λογικός	Περιλαμβάνει δύο τιμές: ΑΛΗΘΗΣ, ΨΕΥΔΗΣ. Χρησιμοποιούνται κυρίως σε λογικές συνθήκες (των δομών επιλογής, επανάληψης) καθώς και ως μεταβλητές-σημαίες . ³	ΑΛΗΘΗΣ, ΨΕΥΔΗΣ. Μία χαρακτηριστική περίπτωση είναι τα πληροφοριακά μηνύματα που εξάγει ο υπολογιστής.

Όπως θα δούμε παρακάτω, για κάθε μεταβλητή ή σταθερά που χρησιμοποιούμε στο πρόγραμμα, πρέπει να δηλώσουμε τον τύπο δεδομένων της.

Σταθερές

Πρόκειται για **τιμές που δεν αλλάζουν κατά τη διάρκεια εκτέλεσης του προγράμματος**. Π.χ. 0.24, 3, 14, 'Μουρατίδης', 'Παρακαλώ, δώστε μία τιμή'.

² Λέγονται **αλφαριθμητικά** διότι μπορούν να περιέχουν (εντός των απλών εισαγωγικών) **αριθμητικούς χαρακτήρες ή/και αλφαβητικούς χαρακτήρες**, (αλφαβητικά + αριθμητικά = αλφαριθμητικά).

³ Οι λεγόμενες **μεταβλητές-σημαίες** έχουν την ίδια σημασία με τους διακόπτες: Βοηθούν στην ενεργοποίηση/απενεργοποίηση καταστάσεων. Π.χ. μία λογική μεταβλητή με όνομα Έγγαμος παίρνει τιμές ΑΛΗΘΗΣ/ΨΕΥΔΗΣ, δηλώνοντας με αυτόν τον τρόπο μία κατάσταση.

Συμβολικές σταθερές ονομάζουμε εκείνες τις **σταθερές που**, για ευκολότερη χρήση, τις **αντιστοιχίζουμε με ονόματα**.

Π.χ. $\Phi\text{ΠΑ} = 0.24$ $\text{ΠΙ} = 3.14$ $\text{ΕΠΩΝΥΜΟ} = \text{'Μουρατίδης'}$.

Μία σταθερά δηλώνεται στην αρχή του προγράμματος και είναι συγκεκριμένου τύπου δεδομένων (ακέραια, πραγματική κλπ.).

Σύνταξη:

ΣΤΑΘΕΡΕΣ

Όνομα-1 = σταθερή-τιμή-1

Όνομα-2 = σταθερά-τιμή-2

...

Όνομα-ν = σταθερά-τιμή-ν

Παραδείγματα:

ΣΤΑΘΕΡΕΣ

$\Phi\text{ΠΑ} = 0.24$

$\text{ΠΙ} = 3.14$

$\text{ΕΠΩΝΥΜΟ} = \text{'Μουρατίδης'}$

Αν αργότερα στο πρόγραμμα, προσπαθήσουμε να αλλάξουμε την τιμή μίας συμβολικής σταθεράς θα είναι **λάθος**:

$\Phi\text{ΠΑ} = 0.23$



Η χρήση των συμβολικών (ή ονοματισμένων) σταθερών κάνει το πρόγραμμα πιο κατανοητό και ευκολότερο να διορθωθεί και συντηρηθεί.

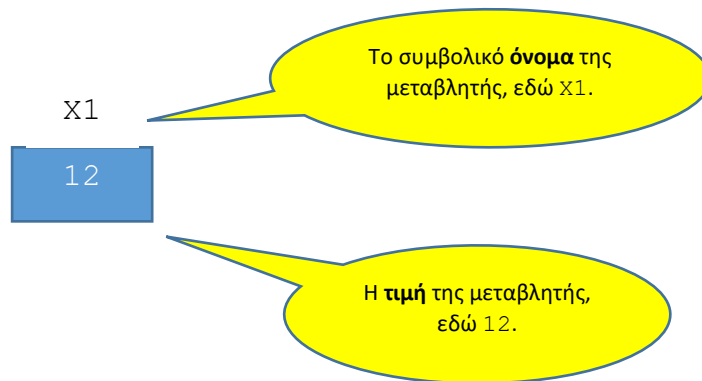
Έτσι, στο παραπάνω παράδειγμα, αν δεν χρησιμοποιούσαμε τη συμβολική σταθερά $\Phi\text{ΠΑ}$ και έπρεπε να αλλάξουμε την τιμή 0.24 σε 0.23 τότε θα έπρεπε να βρούμε όλες τις εμφανίσεις της τιμής 0.24 στο πρόγραμμα, κάτι επίπονο. **Με τη χρήση της συμβολικής σταθεράς $\Phi\text{ΠΑ}$ κάνουμε την αλλαγή μόνο στην αρχή, στο τμήμα δήλωσης σταθερών, το οποίο τμήμα προσδιορίζεται με τη λέξη-κλειδί **ΣΤΑΘΕΡΕΣ**.**

Μεταβλητές

Μία μεταβλητή αναπαριστά μία θέση ή πολλές θέσεις μνήμης⁴ στην οποία αποθηκεύεται μία τιμή. Η θέση ή οι θέσεις μνήμης καθορίζονται από τον μεταγλωττιστή της γλώσσας προγραμματισμού.

Μία μεταβλητή πρέπει να έχει ένα συμβολικό όνομα ώστε να αναφερόμαστε σε αυτήν μέσα στο πρόγραμμα και δηλώνεται στο τμήμα δήλωσης των μεταβλητών που προσδιορίζεται με τη λέξη-κλειδί **ΜΕΤΑΒΛΗΤΕΣ**. Επίσης, **πρέπει να δηλώσουμε και τον τύπο δεδομένων**, για παράδειγμα, ακέραια, πραγματική κλπ.

Σχηματικά:



Σύνταξη:

ΜΕΤΑΒΛΗΤΕΣ

τύπος-1: Λίστα-μεταβλητών-1

τύπος-2: Λίστα-μεταβλητών-2

...

τύπος-ν: Λίστα-μεταβλητών-ν

Για παράδειγμα,

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: X1, X2, Άθροισμα, Π[1:20]

ΠΡΑΓΜΑΤΙΚΕΣ: Μέσος_Όρος, Εμβαδόν

ΧΑΡΑΚΤΗΡΕΣ: Όνομα, Επώνυμο, Διεύθυνση

ΛΟΓΙΚΕΣ: Έγγαμος, Απολύθηκε, Πέρασε_το_μάθημα

⁴ Για παράδειγμα, μία μεταβλητή που αναφέρεται σε έναν πίνακα.



Τα ονόματα των μεταβλητών και συμβολικών σταθερών πρέπει να περιέχουν γράμματα, κεφαλαία ή πεζά του Ελληνικού ή Λατινικού αλφαβήτου, ψηφία (0-9) καθώς και τον χαρακτήρα κάτω παύλα/underscore (_).

Το όνομα πρέπει υποχρεωτικά να ξεκινάει με γράμμα και να μην έχει κενά. Για παράδειγμα, το όνομα 15ΤΙΜΕΣ όπως και το Τιμή προϊόντος δεν είναι έγκυρα.

Δεν μπορούμε, ως όνομα μεταβλητής ή σταθεράς, να χρησιμοποιήσουμε τις ειδικές λέξεις της ΓΛΩΣΣΑΣ (π.χ. ΠΡΟΓΡΑΜΜΑ, ΣΤΑΘΕΡΕΣ, ΜΕΤΑΒΛΗΤΕΣ, ΑΡΧΗ κλπ.) Αυτές οι ειδικές λέξεις είναι δεσμευμένες.



Το όνομα και ο τύπος δεδομένων της μεταβλητής δεν αλλάζουν κατά την εκτέλεση του προγράμματος. Προφανώς όμως, αλλάζει η τιμή της (μεταβάλλεται, εξού και ο όρος μεταβλητή).

Μία καλή πρακτική είναι να χρησιμοποιούμε περιγραφικά ονόματα στις μεταβλητές, δηλαδή ονόματα που να περιγράφουν ή υποδηλώνουν το περιεχόμενό της. Για παράδειγμα, αν έχουμε μία μεταβλητή που αποθηκεύει κάποιον μέσο όρο τότε θα μπορούσαμε να δηλώσουμε το όνομα ΜΟ αλλά ακόμα καλύτερα το όνομα Μέσος_όρος ή ΜέσοςΌρος. Το ίδιο ακολουθούμε και με τις συμβολικές σταθερές που είδαμε προηγουμένως.

Αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές της ΓΛΩΣΣΑΣ είναι οι εξής:

Αριθμητικός τελεστής	Πράξη	Παράδειγμα
+	Πρόσθεση	$x1 + 2$, $x1 + x2$
-	Αφαίρεση	$x1 - 5$, $x1 - x2$

*	Πολλαπλασιασμός	$Y * 5, X * Y$
/	Διαίρεση	$X / 5, X / Y$
^	Ύψωση σε δύναμη	$X ^ 2, X ^ 3$
DIV	Ακέραια διαίρεση	$X \text{ DIV } 5$
MOD	Υπόλοιπο ακέραιας διαίρεσης	$X \text{ MOD } 5$

Ιδιαίτερη προσοχή χρειάζεται στην διαίρεση, διότι όπως παρατηρούμε υπάρχει και η κλασική διαίρεση (/), η ακέραια διαίρεση (DIV) και το υπόλοιπο της ακέραιας διαίρεσης (MOD).

Για **παράδειγμα**, αν η X περιέχει την τιμή 22 τότε:

- $X / 5$ δίνει ως αποτέλεσμα 4.4 (το πηλίκο, μπορεί να είναι πραγματική ή ακέραια τιμή).
- $X \text{ DIV } 5$ δίνει ως αποτέλεσμα 4 (το ακέραιο πηλίκο μόνο).
- $X \text{ MOD } 5$ δίνει ως αποτέλεσμα 2 (το υπόλοιπο της ακέραιας διαίρεσης μόνο).

Συναρτήσεις

Γνωστές συναρτήσεις από τα μαθηματικά μπορούν να χρησιμοποιηθούν και εδώ στη ΓΛΩΣΣΑ ως εξής:

Συνάρτηση	Τι κάνει	Παράδειγμα (θεωρούμε σχεδόν σε όλα $X = 16$)
HM (X)	Υπολογίζει το ημίτονο	HM(16) δίνει 0.276
ΣΥΝ (X)	Υπολογίζει το συνημίτονο	ΣΥΝ(16) δίνει 0.961
ΕΦ (X)	Υπολογίζει την εφαπτομένη	ΕΦ(16) δίνει 0.287

T_P (X)	Υπολογίζει την τετραγωνική ρίζα	T_P(16) δίνει 4
ΛΟΓ (X)	Υπολογίζει τον φυσικό λογάριθμο	ΛΟΓ(16) δίνει 2.77
E (X)	Υπολογίζει το e^x	E(16) δίνει 8.886
A_M (X)	Επιστρέφει το ακέραιο μέρος του X	A_M(16.5) δίνει 16
A_T (X)	Επιστρέφει την απόλυτη τιμή του X	A_T(-16) δίνει 16



Γενικά, οι συναρτήσεις (functions) δέχονται μία ή περισσότερες τιμές ως παραμέτρους, εκτελούν έναν συγκεκριμένο υπολογισμό βάσει αυτών των τιμών-παραμέτρων και επιστρέφουν μία τιμή-αποτέλεσμα.

Οι έτοιμες συναρτήσεις της ΓΛΩΣΣΑΣ, όπως βλέπουμε, δέχονται μία τιμή-παραμέτρο (την συμβολίζουμε X), εκτελείται μία συγκεκριμένη αυτοματοποιημένη επεξεργασία (η οποία δεν μας ενδιαφέρει πώς γίνεται) και επιστρέφει το αποτέλεσμα.

Στο Κεφάλαιο με τα Υποπρογράμματα θα μάθουμε να φτιάχνουμε τις δικές μας συναρτήσεις ανάλογα με την εξειδικευμένη λειτουργία που θέλουμε.

Αριθμητικές εκφράσεις

Μία αριθμητική έκφραση **συνδυάζει αριθμητικές σταθερές, μεταβλητές, συναρτήσεις αριθμητικούς τελεστές και παρενθέσεις**. Οι τελευταίες μάλιστα αλλάζουν και την προτεραιότητα των πράξεων, όπως συμβαίνει και στα μαθηματικά.

Για την **αποτίμηση της τελικής τιμής μίας έκφρασης** παίζει ρόλο η **προτεραιότητα (ιεραρχία) των πράξεων** καθώς και η **ύπαρξη παρενθέσεων**. Ως γνωστόν, η ύπαρξη των παρενθέσεων μεταβάλλει την προτεραιότητα των πράξεων (προηγούνται).

Προτεραιότητα (ιεραρχία) των πράξεων

Όταν έχουμε μία σύνθετη αριθμητική έκφραση που αποτελείται από πολλές πράξεις τότε εκτελούνται με βάση μία προτεραιότητα:

1. Ύψωση σε δύναμη
2. Πολλαπλασιασμός και διαίρεση
3. Πρόσθεση και αφαίρεση

Όταν η ιεραρχία είναι ίδια, δηλαδή έχουμε π.χ. 2 προσθέσεις τότε ο υπολογισμός γίνεται από αριστερά προς δεξιά.

Παραδείγματα:

Μαθηματικά	ΓΛΩΣΣΑ
$X + 1$	$X + 1$
$(X + 1) / 2$	$(X + 1) / 2$
$1/2 \cdot Y^2$	$1 / 2 * Y ^ 2$
$\frac{8X - 5Y}{4}$	$(8 * X - 5 * Y) / 4$
$16 X + 5\eta\mu(X)$	$16 * A_T(X) + 5 * HM(X)$



Όταν γράφουμε μία έκφραση που περιέχει μεταβλητές, αυτές πρέπει να έχουν προηγουμένως μία τιμή. Η απόδοση τιμής σε μία μεταβλητή γίνεται μέσω μίας εντολής εκχώρησης ή μέσω μίας εντολής εισόδου.

Εντολή εκχώρησης

Η απόδοση τιμής σε μία μεταβλητή γίνεται με μία εντολή εκχώρησης.

Σύνταξη:

Εκχώρηση

Όνομα_μεταβλητής ← έκφραση

Το σύμβολο ← υποδηλώνει την τοποθέτηση (εκχώρηση-απόδοση) της τιμής της έκφρασης που είναι δεξιά στην μεταβλητή που είναι αριστερά.⁵

Παραδείγματα:

X ← 5

ΦΠΑ ← 0.19

Y ← 5 * X + 2

ΕΠΩΝΥΜΟ ← 'Μουρατίδης'

ΕΓΓΑΜΟΣ ← ΑΛΗΘΗΣ

Κάτι που πρέπει να προσέξουμε είναι ότι η τιμή της έκφρασης πρέπει να είναι του ίδιου τύπου με την μεταβλητή. Για παράδειγμα, αν η τιμή της έκφρασης είναι πραγματικός αριθμός τότε η μεταβλητή πρέπει να είναι επίσης πραγματικού τύπου.

Για παράδειγμα, το παρακάτω είναι λάθος:

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΑΘΡΟΙΣΜΑ

ΠΡΑΓΜΑΤΙΚΕΣ: X1, X2

Και αργότερα στο πρόγραμμα γράφουμε:

X1 ← 2.5

X2 ← 5.0

⁵ Στις περισσότερες γλώσσες προγραμματισμού το σύμβολο της εκχώρησης είναι το = (ίσον). Αυτό δεν σημαίνει ισότητα αλλά λειτουργεί όπως το σύμβολο ← της ΓΛΩΣΣΑΣ. Π.χ. στην Visual Basic γράφουμε Y = 5 * X + 2 αντί Y ← 5 * X + 2. Σε κάθε περίπτωση, η τιμή της έκφρασης δεξιά εκχωρείται στην μεταβλητή αριστερά.

ΑΘΡΟΙΣΜΑ $\leftarrow X1 + X2$

Η πρόσθεση των $X1$ και $X2$ θα δώσει δεκαδικό αριθμό (δηλαδή, πραγματικού τύπου) και πάει να εκχωρηθεί σε μία μεταβλητή ακέραιου τύπου, κάτι που δεν μπορεί να γίνει (ασυμφωνία τύπων δεδομένων).

Εντολές εισόδου-εξόδου

Κατά τη διάρκεια της εκτέλεσης του προγράμματος ενδέχεται να χρειαστεί ο χρήστης να εισάγει κάποια δεδομένα από το πληκτρολόγιο καθώς και να πραγματοποιηθεί έξοδος κάποιου αποτελέσματος στην οθόνη.

Η ΓΛΩΣΣΑ παρέχει αντίστοιχα τις εντολές ΔΙΑΒΑΣΕ (εντολή εισόδου) και ΓΡΑΨΕ (εντολή εξόδου).

Σύνταξη:

Εντολή εισόδου

ΔΙΑΒΑΣΕ μεταβλητή1, μεταβλητή2, ...

Ο υπολογιστής διακόπτει την εκτέλεση του προγράμματος και περιμένει από τον χρήστη να δώσει τόσες τιμές όσα είναι και τα ονόματα των μεταβλητών δίπλα στην εντολή⁶. Οι τιμές αυτές εκχωρούνται στις αντίστοιχες μεταβλητές.

Παράδειγμα:

ΔΙΑΒΑΣΕ $X1$, $X2$

Ο υπολογιστής διακόπτει την εκτέλεση του προγράμματος και περιμένει από τον χρήστη δύο τιμές (ας υποθέσουμε ακέραιες). Αν δώσει τις τιμές π.χ. 5 και 12 τότε αυτές θα εκχωρηθούν αυτόματα στις μεταβλητές $X1$ και $X2$, αντίστοιχα.



Οι εντολές εισόδου είναι πολύ σημαντικές στον προγραμματισμό, μιας και πετυχαίνουμε την **δυναμική εκχώρηση τιμών στις μεταβλητές**. Δηλαδή, κάθε φορά που εκτελείται το πρόγραμμα, ο χρήστης μπορεί να δίνει διαφορετικές τιμές εισόδου προς επεξεργασία.

⁶ Η λίστα των μεταβλητών δίπλα στην εντολή ΔΙΑΒΑΣΕ διαχωρίζεται με το , (κόμμα).

Σύνταξη:

Εντολή εξόδου

ΓΡΑΨΕ λίστα_στοιχείων

Η λίστα_στοιχείων μπορεί να είναι οποιοσδήποτε συνδυασμός από σταθερές τιμές (συνήθως περιγραφικές λέξεις ή φράσεις) και ονόματα μεταβλητών⁷. Το τελικό αποτέλεσμα είναι μία συνένωση όλων αυτών (σταθερών τιμών και τιμών των μεταβλητών) σε ένα αποτέλεσμα που εμφανίζεται επί της οθόνης.

Παραδείγματα:

ΓΡΑΨΕ 'Ο μέσος όρος είναι ' , Μέσος_όρος



Αν η μεταβλητή Μέσος_όρος έχει την τιμή 12.5, τότε το μήνυμα επί της οθόνης που θα εμφανιστεί θα είναι «Ο μέσος όρος είναι 12.5».

ΓΡΑΨΕ 'Το άθροισμα είναι ' , SUM , ' και μέσος όρος είναι ' , AVERAGE

Εδώ, συνενώνονται 4 στοιχεία (δύο σταθερές τιμές-λεκτικά και δύο μεταβλητές). Αν η SUM έχει την τιμή 25 και η AVERAGE την τιμή 3.4 τότε το μήνυμα διαμορφώνεται ως εξής:

«Το άθροισμα είναι 25 και ο μέσος όρος είναι 3.4»



Οι εντολές εξόδου χρησιμοποιούνται κυρίως για την εμφάνιση κατάλληλων μηνυμάτων (πχ. 'Παρακαλώ δώστε μία τιμή') καθώς και των αποτελεσμάτων επεξεργασίας που είναι αποθηκευμένα σε μεταβλητές (πχ. ένα άθροισμα, ένας μέσος όρος).

⁷ Τα στοιχεία (σταθερές και ονόματα μεταβλητών) στην εντολή ΓΡΑΨΕ διαχωρίζονται με το , (κόμμα).

Δομή προγράμματος

Η κατασκευή του προγράμματος έχει μία συγκεκριμένη δόμηση (δηλαδή, ακολουθεί και αυτή κάποιους συγκεκριμένους συντακτικούς αυστηρούς κανόνες), η οποία είναι η εξής:

ΠΡΟΓΡΑΜΜΑ όνομα_προγράμματος

Επικεφαλίδα

ΣΤΑΘΕΡΕΣ

Δήλωση συμβολικών σταθερών

Τμήμα δήλωσης σταθερών

ΜΕΤΑΒΛΗΤΕΣ

Δήλωση μεταβλητών

Τμήμα δήλωσης μεταβλητών

ΑΡΧΗ

Εντολές

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Κύριο τμήμα του προγράμματος

Παρατηρήσεις:

- Αν δεν χρησιμοποιούμε συμβολικές σταθερές στο πρόγραμμά μας, τότε το τμήμα δήλωσης σταθερών παραλείπεται.
- Στο κύριο τμήμα του προγράμματος (**ΑΡΧΗ ... ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ**), όσες μεταβλητές χρησιμοποιούμε πρέπει να τις έχουμε ήδη δηλώσει στο τμήμα δήλωσης μεταβλητών (**ΜΕΤΑΒΛΗΤΕΣ**).
- Κάθε εντολή γράφεται σε ξεχωριστή γραμμή. Αν μία εντολή πρέπει να συνεχιστεί και στην επόμενη γραμμή, τότε ο πρώτος χαρακτήρας αυτής της γραμμής πρέπει να είναι ο χαρακτήρας & (αγγλικό και).
- Αν θέλουμε να γράψουμε ένα σχόλιο σε μία γραμμή, τότε ο πρώτος χαρακτήρας πρέπει να είναι το θαυμαστικό (!). Ο υπολογιστής αγνοεί τις γραμμές σχολίων διότι δεν περιέχουν εκτελέσιμες εντολές.

Παραδείγματα προγραμμάτων στη ΓΛΩΣΣΑ

Παράδειγμα 1

Να γραφεί πρόγραμμα στη ΓΛΩΣΣΑ που επιλύει το πρόβλημα του υπολογισμού του εμβαδού ενός τριγώνου. Το πρόγραμμα θα δέχεται ως δεδομένα τη βάση και το ύψος του τριγώνου, θα υπολογίζει το εμβαδόν και θα τυπώνει το αποτέλεσμα στην οθόνη.

ΠΡΟΓΡΑΜΜΑ Εμβαδόν_τριγώνου

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: Βάση, Ύψος, Εμβαδόν

ΑΡΧΗ

! Εισαγωγή των δεδομένων από το πληκτρολόγιο, μετά από ένα ευγενικό μήνυμα του υπολογιστή.

ΓΡΑΨΕ 'Παρακαλώ, δώστε την βάση'

ΔΙΑΒΑΣΕ Βάση

ΓΡΑΨΕ 'Παρακαλώ, δώστε το ύψος'

ΔΙΑΒΑΣΕ Ύψος

! Επεξεργασία των δεδομένων, εδώ η εφαρμογή του γνωστού τύπου.

! Μετά τον υπολογισμό της έκφρασης (βάση x ύψος) / 2, η τιμή του αποτελέσματος

! εκχωρείται στην μεταβλητή Εμβαδόν.

Εμβαδόν ← (Βάση * Ύψος) / 2

! Εμφάνιση του αποτελέσματος (έξοδος στην οθόνη).

ΓΡΑΨΕ 'Το εμβαδόν του τριγώνου είναι ', Εμβαδόν

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Ας δούμε τη **ροή εκτέλεσης των εντολών**:

1. Ο υπολογιστής **πρώτα εκτελεί την εντολή ΓΡΑΨΕ** όπου εμφανίζει ένα ευγενικό μήνυμα προς τον χρήστη και τον παρακαλεί να δώσει τη βάση του τριγώνου.
2. Στη συνέχεια, **εκτελεί την εντολή ΔΙΑΒΑΣΕ** όπου διακόπτει την εκτέλεση του προγράμματος και περιμένει τον χρήστη να πληκτρολογήσει το δεδομένο. Ο χρήστης, ως υποθέσουμε ότι πληκτρολογεί τον αριθμό 12.5 και πατάει Enter. Τότε ο υπολογιστής παίρνει τον αριθμό 12.5 και τον εκχωρεί στην μεταβλητή Βάση. Αν ο χρήστης δεν είχε δώσει αριθμό θα προέκυπτε σφάλμα⁸.
3. Στη συνέχεια, **εκτελεί την επόμενη εντολή ΓΡΑΨΕ** όπου εμφανίζει ένα δεύτερο ευγενικό μήνυμα προς τον χρήστη και τον παρακαλεί να δώσει το ύψος του τριγώνου.
4. Στη συνέχεια, **εκτελεί την επόμενη εντολή ΔΙΑΒΑΣΕ** όπου διακόπτει την εκτέλεση του προγράμματος και περιμένει τον χρήστη να πληκτρολογήσει το δεδομένο. Ο χρήστης, ως υποθέσουμε ότι πληκτρολογεί τον αριθμό 4 και πατάει Enter. Τότε ο υπολογιστής παίρνει τον αριθμό 4 και τον εκχωρεί στην μεταβλητή Ύψος.
5. Στη συνέχεια, **κάνει τον υπολογισμό του εμβαδού εκτελώντας την αντίστοιχη εντολή εκχώρησης**. Υπολογίζει την έκφραση στο δεξί μέρος $(12.5 * 4 / 2)$ και η τιμή του αποτελέσματος (25) αποδίδεται-εκχωρείται στην μεταβλητή Εμβαδόν.
6. Τέλος, **εκτελεί την τελευταία εντολή ΓΡΑΨΕ** όπου συνενώνει μία σταθερή τιμή (το λεκτικό της περιγραφής) με την τιμή της μεταβλητής Εμβαδόν. Το μήνυμα «Το εμβαδόν του τριγώνου είναι 25» εμφανίζεται επί της οθόνης.



Το παραπάνω, αποτελεί ένα απλό κλασικό παράδειγμα. Περιλαμβάνει **τρία τμήματα** (με τα ανάλογα πράσινα σχόλια). Η **είσοδος των δεδομένων**, η **επεξεργασία** τους και η **έξοδος του/των αποτελέσματος/των**. Αυτά τα τμήματα μπορεί, σε πιο σύνθετα προβλήματα, να έχουν διακλαδώσεις στη ροή εκτέλεσης (με εντολές ΑΝ ή επαναληπτικές). Ανεξάρτητα από την πολυπλοκότητα, σε γενικές γραμμές, η πλειονότητα των προγραμμάτων ακολουθεί αυτήν την τμηματοποίηση (Είσοδος-Επεξεργασία-Έξοδος).

⁸ Πάντα **κατά την εισαγωγή των δεδομένων γίνεται έλεγχος εγκυρότητας**. Δηλαδή, ελέγχει αν το εισαγόμενο δεδομένο συμφωνεί με τον τύπο δεδομένων της μεταβλητής στην οποία θα εκχωρηθεί. Αν, για παράδειγμα, δοθεί ο αριθμός 12.5 προς εκχώρηση σε μία ακεραίου τύπου μεταβλητή θα προκληθεί **σφάλμα εκτέλεσης**. Τα προγράμματα ΠΡΕΠΕΙ να περιλαμβάνουν κώδικα ελέγχου εγκυρότητας των εισαγόμενων δεδομένων.

Παράδειγμα 2

Να γραφεί πρόγραμμα στη ΓΛΩΣΣΑ που διαβάζει τους βαθμούς των δύο τετραμήνων και του γραπτού ενός μαθητή στο μάθημα της Πληροφορικής και υπολογίζει και τυπώνει τον μέσο όρο, για αυτό το μάθημα.

ΠΡΟΓΡΑΜΜΑ Μέσος_όρος_μαθήματος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: T1, T2, Γ

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΑΡΧΗ

! Εισαγωγή των δεδομένων από το πληκτρολόγιο, μετά από ένα ευγενικό μήνυμα του υπολογιστή.

ΓΡΑΨΕ 'Παρακαλώ, δώστε τον βαθμό του 1^{ου} τετραμήνου'

ΔΙΑΒΑΣΕ T1

ΓΡΑΨΕ 'Παρακαλώ, δώστε τον βαθμό του 2^{ου} τετραμήνου'

ΔΙΑΒΑΣΕ T2

ΓΡΑΨΕ 'Παρακαλώ, δώστε τον βαθμό των Γραπτών'

ΔΙΑΒΑΣΕ Γ

! Επεξεργασία των δεδομένων.

$ΜΟ \leftarrow (T1 + T2 + Γ) / 3$

! Εμφάνιση του αποτελέσματος (έξοδος στην οθόνη).

ΓΡΑΨΕ 'Ο μέσος όρος του μαθήματος είναι ' , ΜΟ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Φυσικά, μία καλή τακτική είναι να δίνουμε πιο περιγραφικά ονόματα στις μεταβλητές. Αντί του T1 θα ήταν προτιμητέο το Τετράμηνο_1 ή Τετράμηνο1.

Παράδειγμα 3

Ένας καταστηματάρχης θέλει να υπολογίσει την τιμή ενός προϊόντος με δύο διαφορετικούς συντελεστές έκπτωσης. Να γραφεί πρόγραμμα στη ΓΛΩΣΣΑ που διαβάζει το όνομα του προϊόντος, την κανονική τιμή του (προ έκπτωσης) και υπολογίζει την τιμή του προϊόντος με έκπτωση 15% και την τιμή του προϊόντος με έκπτωση 20%.

ΠΡΟΓΡΑΜΜΑ Έκπτωση_προϊόντος

ΣΤΑΘΕΡΕΣ

! Συντελεστές έκπτωσης.

Έκπτωση1 = 0.15

Έκπτωση2 = 0.20

ΜΕΤΑΒΛΗΤΕΣ

ΧΑΡΑΚΤΗΡΕΣ: Όνομα_προϊόντος

ΠΡΑΓΜΑΤΙΚΕΣ: Τιμή, Τιμή_με_έκπτωση1, Τιμή_με_έκπτωση2

ΑΡΧΗ

! Εισαγωγή των δεδομένων από το πληκτρολόγιο, μετά από ένα ευγενικό μήνυμα του υπολογιστή.

ΓΡΑΨΕ 'Παρακαλώ, δώστε τον όνομα του προϊόντος'

ΔΙΑΒΑΣΕ Όνομα_προϊόντος

ΓΡΑΨΕ 'Παρακαλώ, δώστε την κανονική (προ έκπτωσης) τιμή'

ΔΙΑΒΑΣΕ Τιμή

! Επεξεργασία των δεδομένων.

Τιμή_με_έκπτωση1 ← Τιμή * (1 - Έκπτωση1)

Τιμή_με_έκπτωση2 ← Τιμή * (1 - Έκπτωση2)

! Εμφάνιση των αποτελεσμάτων (έξοδος στην οθόνη).

ΓΡΑΨΕ 'Το προϊόν ` , Όνομα_προϊόντος, ` έχει την κανονική & τιμή ` , Τιμή, ` με τις εξής εκπτώσεις: `

ΓΡΑΨΕ 'Με έκπτωση 15% η τιμή διαμορφώνεται σε ` , & Τιμή_με_έκπτωση1

ΓΡΑΨΕ 'Με έκπτωση 20% η τιμή διαμορφώνεται σε ` , & Τιμή_με_έκπτωση2

ΤΕΛΟΣ ΠΡΟΓΡΑΜΜΑΤΟΣ

Ερωτήσεις κατανόησης

1. Τί σημαίνει ο όρος «**τύπος δεδομένων**» στον προγραμματισμό;
2. Αναφέρατε τους τύπος δεδομένων που χρησιμοποιεί η ΓΛΩΣΣΑ. Δώστε κι από 2 παραδείγματα για κάθε τύπο.
3. Ποιά η χρησιμότητα των **συμβολικών σταθερών** σε ένα πρόγραμμα; Σε ποιά θέση του προγράμματος δηλώνονται;
4. Είναι υποχρεωτικό να χρησιμοποιούμε συμβολικές σταθερές;
5. Ποιά η χρησιμότητα των **μεταβλητών** σε ένα πρόγραμμα και σε τι διαφέρουν από τις συμβολικές σταθερές;
6. Ποιά από τα παρακάτω ονόματα μεταβλητών **δεν είναι έγκυρα** στη ΓΛΩΣΣΑ:
 - a. Τιμή_προϊόντος
 - b. ΤιμήΠροϊόντος
 - c. _Τιμή_προϊόντος
 - d. Τιμή προϊόντος
 - e. 15ΑΒΓ
 - f. Τιμή_προϊόντος\$
 - g. %Τιμή_προϊόντος
7. Όταν δηλώνουμε τις μεταβλητές πρέπει υποχρεωτικά να καθορίσουμε και τον τύπο τους;
8. Μετατρέψτε τις παρακάτω περιγραφές σε εντολές της ΓΛΩΣΣΑΣ:
 - a. Τοποθετώ την τιμή 15 στην μεταβλητή X.
 - b. Τοποθετώ την λέξη «Καλωσήρθατε» στην μεταβλητή Message.
 - c. Τοποθετώ την τιμή ΑΛΗΘΗΣ στην μεταβλητή Pass.
 - d. Τοποθετώ το γινόμενο των μεταβλητών X και Y στην μεταβλητή Z.
 - e. Τοποθετώ το υπόλοιπο της διαίρεσης των μεταβλητών X και Y στην μεταβλητή Z.
 - f. Τοποθετώ την τετραγωνική ρίζα της μεταβλητής X στην μεταβλητή Y.

- g. Τοποθετώ στην μεταβλητή Y το άθροισμα των απόλυτων τιμών των μεταβλητών X1 και X2.
9. Δίνεται η εξής περιγραφή: «Αφού αθροίσω τις μεταβλητές X1, X2 και X3, στη συνέχεια διαιρώ με το 25. Κατόπιν αθροίζω με το ημίτονο της X1 και όλο αυτό διαιρείται με το 100».
Μετατρέψτε την άνω περιγραφή σε αριθμητική έκφραση.
10. Σε τί χρησιμεύει η εντολή εισόδου ΔΙΑΒΑΣΕ της ΓΛΩΣΣΑΣ;
11. Σε τί χρησιμεύει η εντολή εξόδου ΓΡΑΨΕ της ΓΛΩΣΣΑΣ;
12. Περιγράψτε τη δομή ενός προγράμματος στη ΓΛΩΣΣΑ.
13. Τί υποδηλώνει η φράση «ροή εκτέλεσης» σε ένα πρόγραμμα;
14. Εντοπίστε τα συντακτικά λάθη στο παρακάτω πρόγραμμα της ΓΛΩΣΣΑΣ:

```
ΠΡΟΓΡΑΜΜΑ Εμβαδόν κύκλου
ΣΤΑΘΕΡΑ
    ΠΙ = 3.14
ΜΕΤΑΒΛΗΤΕΣ
    ΠΡΑΓΜΑΤΙΚΗ: Ακτίνα
ΑΡΧΗ
    ! Εισαγωγή των δεδομένων από το πληκτρολόγιο, μετά από ένα ευγενικό
    ! μήνυμα του υπολογιστή.
    ΓΡΑΨΕ 'Παρακαλώ, δώστε την ακτίνα του κύκλου'
    ΔΙΑΒΑΣΕ Ακτίνα κύκλου
    ! Επεξεργασία των δεδομένων.
    Εμβαδόν ← (ΠΙ * Ακτίνα ^ 2)
    ! Εμφάνιση του αποτελέσματος (έξοδος στην οθόνη).
    ΕΚΤΥΠΩΣΕ 'Το εμβαδόν του κύκλου είναι ', Εμβαδόν
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

Προσπαθήστε μόνοι σας πριν δείτε την απάντηση παρακάτω.

Απάντηση στην ερώτηση 14:

- A. Το όνομα του προγράμματος έχει κενό. Το σωστό είναι Εμβαδόν_κύκλου ή ΕμβαδόνΚύκλου.
- B. Η λέξη ΣΤΑΘΕΡΑ δεν υπάρχει στο λεξιλόγιο της ΓΛΩΣΣΑΣ. Το σωστό είναι ΣΤΑΘΕΡΕΣ.
- C. Η λέξη ΠΡΑΓΜΑΤΙΚΗ δεν υπάρχει στο λεξιλόγιο της ΓΛΩΣΣΑΣ. Το σωστό είναι ΠΡΑΓΜΑΤΙΚΕΣ.
- D. Στην εντολή ΔΙΑΒΑΣΕ υπάρχει διπλό λάθος: Το όνομα της μεταβλητής Ακτίνα κύκλου έχει κενό κι επιπλέον δεν έχει δηλωθεί στο τμήμα ΜΕΤΑΒΛΗΤΕΣ. Το σωστό είναι Ακτ ίνα.
- E. Δεν έχει δηλωθεί στο τμήμα ΜΕΤΑΒΛΗΤΕΣ η μεταβλητή Εμβαδόν.
- F. Η εντολή ΕΚΤΥΠΩΣΕ δεν υπάρχει στο ρεπερτόριο εντολών της ΓΛΩΣΣΑΣ. Το σωστό είναι ΓΡΑΨΕ.

Πολλά λάθη για ένα τόσο μικρό πρόγραμμα!

ΤΕΛΟΣ ΣΗΜΕΙΩΣΕΩΝ ΚΕΦΑΛΑΙΟΥ 7