# Προγραμματισμός Υπολογιστών

**Κεφάλαιο 8  Η γλώσσα Pascal**

# Παράγραφος 8.2

## Βασικοί τύποι δεδομένων

## 8.2. Βασικοί τύποι δεδομένων

- Τα **δεδομένα** ενός προγράμματος μπορεί να:
  - είναι αποθηκευμένα εσωτερικά στην μνήμη
  - είναι αποθηκευμένα εξωτερικά στο δίσκο
  - εισάγονται από το πληκτρολόγιο ή μια συσκευή εισόδου (π.χ. σαρωτής)
- Καθένα δεδομένο **πρέπει** να είναι ενός ορισμένου τύπου
- **Τύπος δεδομένων (Data type)**
  - Μια *κατηγορία* δεδομένων με ορισμένη *απεικόνιση* και ένα *σύνολο λειτουργιών* που μπορούν να εφαρμοστούν στο σύνολο των τιμών τους

## 8.2. Βασικοί τύποι δεδομένων

- Προσδιορίζουν:
  - τον τρόπο **παράστασης** των δεδομένων εσωτερικά στον υπολογιστή καθώς και
  - το είδος της **επεξεργασίας** τους από τον υπολογιστή
- Είδη:
  - Προσδιορισμένοι απ’ την γλώσσα (build-in)
  - Καθορισμένοι από το χρήστη (user-defined)
- Χρησιμοποιούνται για τη δήλωση των <u>μεταβλητών</u> ή των <u>συναρτήσεων</u> που ορίζει ο χρήστης
- **Μια μεταβλητή είναι πάντοτε ενός και μόνο συγκεκριμένου τύπου**

## 8.2. Βασικοί τύποι δεδομένων

- Κατηγορίες:
  - **απλοί** τύποι ή στοιχειώδεις
  - **σύνθετοι** τύποι
- Οι προσδιορισμένοι από την Pascal απλοί τύποι είναι:
  - Ο **ακέραιος** τύπος (Integer)
  - Ο **πραγματικός** τύπος (Real)
  - Ο **λογικός** τύπος (Boolean)
  - Ο **χαρακτήρας** (Char)
- Σύνθετοι τύποι είναι αυτοί που ορίζονται από απλούς τύπους ή από άλλους σύνθετους που ορίστηκαν παραπάνω. Παράδειγμα σύνθετου τύπου είναι ο **αλφαριθμητικός** τύπος (String)

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.1. Ακέραιος (Integer)*

- Μπορεί να είναι θετικοί ή αρνητικοί αριθμοί
- Το πλήθος των ψηφίων περιορίζεται από τον τύπο του υπολογιστή
- Η μεταβλητή `MaxInt` προσδιορίζει το εύρος του διαστήματος των ακεραίων το οποίο είναι από `–MaxInt–1` έως `MaxInt` δηλαδή στο διάστημα [-32768, 32767]

# 8.2. Βασικοί τύποι δεδομένων

## *8.2.1. Ακέραιος (Integer)*

- Η γλώσσα Pascal υποστηρίζει διάφορους τύπους ακεραίων ανάλογα με:
  - το πεδίο τιμών τους
  - την ύπαρξη ή μη πρόσημου
  - το πλήθος των bytes που καταλαμβάνουν στην κεντρική μνήμη

| Τύπος | Διάστημα τιμών | Πρόσημο | Bytes |
|---|---|---|---|
| ShortInt | -128 .. 127 | ΝΑΙ | 1 |
| Integer | -32768 .. 32767 | ΝΑΙ | 2 |
| LongInt | -2148483648 .. 2148483647 | ΝΑΙ | 4 |
| Byte | 0 .. 255 | ΟΧΙ | 1 |
| Word | 0 .. 65535 | ΟΧΙ | 2 |

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.1. Ακέραιος (Integer)*

- Οι επιτρεπτές πράξεις μεταξύ ακεραίων είναι:
  - Πρόσθεση **+**
  - Αφαίρεση **-**
  - Πολλαπλασιασμός *****
  - Ακέραια διαίρεση (πηλίκο) **div**
  - Υπόλοιπο διαίρεσης **mod**

<u>Παραδείγματα</u>:

```
27 div 6 = 4,      27 mod 6 = 3
16 div 17 = 0,     16 mod 17 = 16
36 div 6 = 6,      36 mod 6 = 0
```

## *8.2.2. Πραγματικός (Real)*

- Χρησιμοποιείται όταν:
  - Οι αριθμητικές τιμές δεν είναι ακέραιες (δεκαδικοί αριθμοί)
  - Οι αναμενόμενες τιμές είναι εκτός ορίων του ακεραίου τύπου
  - Θέλουμε να εκφράσουμε πολύ μεγάλους ή πολύ μικρούς αριθμούς
- Οι πραγματικοί τύποι που διαθέτει η Pascal είναι οι:
  - Real (11-12 ψηφία)
  - Single (7-8 ψηφία)
  - Double (15-16 ψηφία)
  - Extended (19-20 ψηφία)
  - Comp (19-20 ψηφία)

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.2. Πραγματικός (Real)*

- Οι επιτρεπτές πράξεις πραγματικών αριθμών είναι:
  - Πρόσθεση **+**
  - Αφαίρεση **-**
  - Πολλαπλασιασμός *****
  - Διαίρεση **/**

Σημείωση:
Πράξεις που περιλαμβάνουν πραγματικούς αριθμούς απαιτούν **περισσότερο χρόνο** για να εκτελεστούν από αυτές που περιλαμβάνουν μόνον ακέραιους.

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.2. Αριθμητικές εκφράσεις*

Είναι απεικονίσεις αριθμητικών παραστάσεων που μπορεί να περιέχουν *σταθερές*, *μεταβλητές*, *συναρτήσεις*, *αριθμητικά σύμβολα* και *παρενθέσεις*.

### *8.2.2. Προτεραιότητα αριθμητικών τελεστών*

**Χαμηλότερη** προτεραιότητα

| | |
|---|---|
| **+** | (πρόσθεση) |
| **-** | (αφαίρεση) |

**Υψηλότερη** προτεραιότητα

| | |
|---|---|
| ***** | (πολλαπλασιασμός) |
| **/** | (διαίρεση) |
| **div** | (ακέραια διαίρεση) |
| **mod** | (υπόλοιπο ακεραίας διαίρεσης) |

### *8.2.2. Προτεραιότητα αριθμητικών τελεστών*

Προτεραιότητα σημαίνει ότι κάθε πράξη με σύμβολα *, **/**, **div**, **mod** εκτελείται πρώτη εκτός αν υπάρχει **παρένθεση**, οπότε εκτελείται πρώτα η πράξη της παρένθεσης π.χ.

```
6 * 3 + 5 = 23
6 * (3 + 5) = 48
```

Όταν οι τελεστές είναι της ίδιας προτεραιότητας οι πράξεις εκτελούνται από τα αριστερά προς τα δεξιά π.χ.

```
20 div 3 * 4 = 24
```
(Η ακέραια διαίρεση και ο πολλαπλασιασμός έχουν την ίδια προτεραιότητα)

```
20 mod 3 * 4 = 8
```
(Το υπόλοιπο της διαίρεσης και ο πολλαπλασιασμός έχουν την ίδια προτεραιότητα)

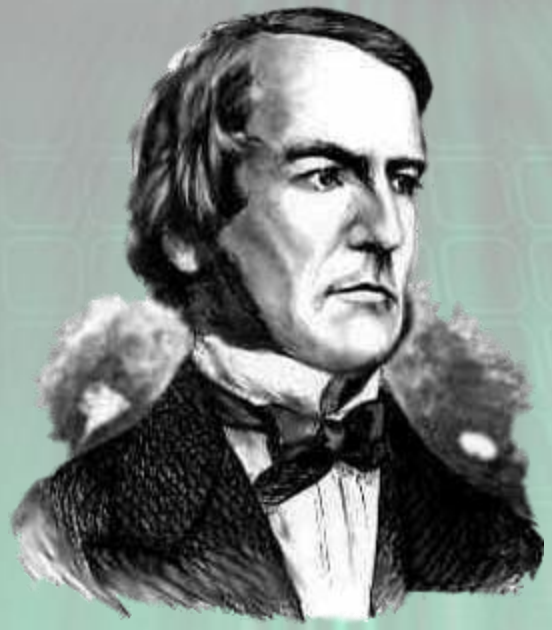
```
6 + 12 / (2 * 3) = 8
```
(Η παρένθεση έχει τη μεγαλύτερη προτεραιότητα και ακολουθεί η διαίρεση)

```
15 / 3 * 2 = 10
```
(Η διαίρεση και ο πολλαπλασιασμός έχουν την ίδια προτεραιότητα)

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.3. Λογικός (Boolean)*

Σκοπός μιας μεταβλητής λογικού τύπου είναι η καταγραφή του αποτελέσματος ενός ελέγχου. Για να εξετάσουμε το αποτέλεσμά του, αρκεί να εξετάσουμε την τιμή της λογικής μεταβλητής.



- Ο λογικός τύπος (Boolean) μπορεί να έχει μία από τις τιμές:
  - **True** (Αληθής)
  - **False** (Ψευδής)

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.3. Λογικές εκφράσεις*

- Είναι απεικονίσεις παραστάσεων που μπορεί να περιέχουν σταθερές, μεταβλητές, συναρτήσεις, αριθμητικά σύμβολα και παρενθέσεις
- **Μπορούν να πάρουν μία από τις δύο λογικές τιμές (True, False)**
- Παράγονται από δύο μεταβλητές, σταθερές ή συναρτήσεις μέσω των σχεσιακών τελεστών

Παραδείγματα:

```
(X > 1)
(X + 1 > 0)
```

# 8.2. Βασικοί τύποι δεδομένων

## *8.2.3. Σχεσιακοί τελεστές*

| Περιγραφή | Μαθηματικά | Pascal |
|---|---|---|
| Ίσο | = | = |
| Μεγαλύτερο από | > | > |
| Μικρότερο από | < | > |
| Μεγαλύτερο ή ίσο | ≥ | >= |
| Μικρότερο ή ίσο | ≤ | <= |
| Διάφορο | ≠ | <> |
| Ανήκει | ∈ | in |

### *8.2.3. Λογικές πράξεις*

- Οι πράξεις που μπορούν να γίνουν με μεταβλητές ή εκφράσεις λογικού τύπου είναι:
  - Η **σύζευξη** (**and**)
  - Η **διάζευξη** (**or**)
  - Η **άρνηση** (**not**)
  - Η **αποκλειστική διάζευξη** (**xor**)

Παραδείγματα:

```
(X > 1) and (X < 10)
((X > 0) and (Y > 0)) or (Z <> 1)
```

## *8.2.3. Πίνακες αλήθειας*

Πίνακες αλήθειας των τεσσάρων λογικών πράξεων μεταξύ δύο λογικών μεταβλητών **P** και **Q**

| P | Q | P and Q | P or Q | not P | P xor Q |
|---|---|---|---|---|---|
| True | True | True | True | False | False |
| True | False | False | True | False | True |
| False | True | False | True | True | True |
| False | False | False | False | True | False |

Σημείωση:
Συχνά αναφερόμαστε στις τέσσερις λογικές πράξεις **and**, **or**, **xor** και **not** με το όνομα **λογικοί τελεστές**

### *8.2.3. Λογικοί τελεστές*

- Ο τελεστής **and** είναι `True` μόνο όταν και οι δύο λογικές εκφράσεις έχουν τιμή `True`
  - Τελεστής λογικού πολλαπλασιασμού
- Ο τελεστής **or** είναι `True` όταν τουλάχιστον μία από τις λογικές εκφράσεις έχει τιμή `True`
  - Τελεστής λογικής πρόσθεσης
- Ο τελεστής **and** έχει μεγαλύτερη προτεραιότητα από τον τελεστή **or**
  - `((x >= 0) and (x <= 10)) or (x = 20)`
  - `(x >= 0) and (x <= 10) or (x = 20)`

### *8.2.3. Λογικοί τελεστές (συνέχεια)*

- Ο τελεστής `xor` είναι `True` όταν μόνο μία από τις δύο λογικές εκφράσεις έχει τιμή `True`
  - Έχει την ίδια προτεραιότητα με τον τελεστή `or`
- Ο τελεστής `not` δέχεται μία λογική έκφραση και της αντιστρέφει την τιμή

# 8.2. Βασικοί τύποι δεδομένων

## *8.2.3. Ευρετήριο τοπικών όρων*

- Αριθμητικές εκφράσεις
  - Αριθμητικοί τελεστές
- Λογικές εκφράσεις
  - Σχεσιακοί ή συγκριτικοί τελεστές
  - Λογικοί τελεστές
- Προτεραιότητα τελεστών
  - Αριθμητικοί > Σχεσιακοί
  - Λογικοί > Σχεσιακοί
  - Αριθμητικοί ? Λογικοί

## 8.2. Βασικοί τύποι δεδομένων

### *8.2.4. Χαρακτήρας (Char)*

- Περιγράφει δεδομένα **ενός** χαρακτήρα στο σύνολο των χαρακτήρων που δύναται να αναπαρασταθούν σε έναν υπολογιστή
- **Διατεταγμένος** τύπος
  - ‘Γ’ < ‘Δ’ < ‘Ε’ και ‘0’ < ‘1’ < ‘2’
  - Η διάταξη διαφέρει από υπολογιστή σε υπολογιστή

### *8.2.5. Αλφαριθμητικός τύπος (string)*

- Δεν συναντάται στην Standard Pascal
- Μια σειρά από 0 έως το πολύ 255 χαρακτήρες
- Δύναται να δηλωθεί το μήκος του αλφαριθμητικού π.χ. Η δήλωση `FName: string[80]` δηλώνει μια μεταβλητή που μπορεί να δεχθεί το πολύ 80 χαρακτήρες
- Ένα string **πρέπει** να γράφεται στην ίδια γραμμή
- `‘Vathmos’ <> Vathmos`
- `‘2468’ <> 2468`
- Κενό string
- Λειτουργίες **συνένωσης**, **απομάκρυνσης** και **σύγκρισης**