



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Ανώτατο Εκπαιδευτικό Ίδρυμα Πειραιά
Τεχνολογικού Τομέα



Εισαγωγή στην Πληροφορική & τον Προγραμματισμό

Ενότητα 8^η: Διαδικασίες (Sub & Function)

Ι. Ψαρομήλιγκος – Χ. Κυτάγιας
Τμήμα Διοίκησης Επιχειρήσεων



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Ανώτατο Εκπαιδευτικό Ίδρυμα Πειραιά Τεχνολογικού Τομέα**» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Σκοποί ενότητας

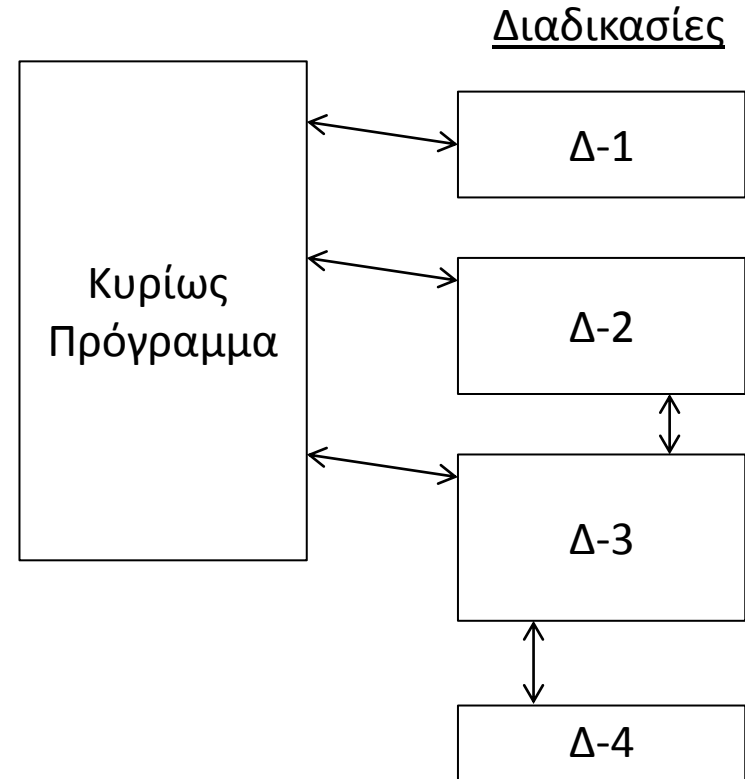
Στην συγκεκριμένη ενότητα παρουσιάζονται οι διαδικασίες τύπου Sub και Function, παράμετροι και επιστρεφόμενες τιμές, πέρασμα παραμέτρων (ByVal, ByVal) καθώς και θέματα εμβέλειας διαδικασιών & μεταβλητών της γλώσσας προγραμματισμού Visual Basic.NET

Περιεχόμενα

- Η Διαδικασία Function
- Η Διαδικασία Sub
- Παράμετροι
- Επιστρεφόμενες τιμές
- Θέματα Εμβέλειας

Διαδικασίες Εισαγωγικά

- Η διαδικασία είναι μια «επώνυμη» μονάδα κώδικα.
- Την «καλούμε» με το όνομά της κάθε φορά που χρειαζόμαστε να εκτελεστεί το σύνολο των εντολών που περιέχει.
- Στη συνέχεια ο έλεγχος μεταφέρεται στο τμήμα του κώδικα που την κάλεσε



Είδη Διαδικασιών

- Μια διαδικασία υλοποιεί μια συγκεκριμένη εργασία, άρα με αυτό τον τρόπο κάνουμε «**οικονομία**» στον κώδικα του προγράμματός μας χωρίζοντάς τον σε μικρότερα τμήματα που τα καλούμε κάθε φορά που τα χρειαζόμαστε (εύκολη συντήρηση του κώδικά μας)
- Η Visual Basic παρέχει δύο ειδών διαδικασίες: **Sub** και **Function**
- Η διαφορά των διαδικασιών τύπου **Function** από τις διαδικασίες τύπου **Sub** βρίσκεται στο γεγονός ότι οι functions έχουν υποχρέωση μετά την κλήση τους να **επιστρέψουν μια τιμή**. Η τιμή αυτή επιστρέφεται στο όνομα της διαδικασίας

Η Διαδικασία Function

- Ορίζεται με δηλωτικές προτάσεις που καθορίζουν την **αρχή** και το **τέλος** της.
- Μεταξύ της αρχής και του τέλους της διαδικασίας τοποθετούνται οι εντολές που θέλουμε να εκτελεστούν
 - οι εντολές μπορεί να είναι εκτελέσιμες προτάσεις που να καλούν άλλες διαδικασίες τύπου Sub ή function καθώς και δηλωτικές προτάσεις
- Στην **αρχική δηλωτική πρόταση** μιας διαδικασίας προσδιορίζονται:
 - η **εμβέλεια (scope)** της, δηλαδή αν θα είναι «ορατή» και θα μπορεί να κληθεί από άλλες ενότητες κώδικα
 - το **όνομά της**, με το οποίο θα αναγνωρίζεται από το σύστημα της γλώσσας προγραμματισμού Visual Basic
 - οι **παράμετροι της**, δηλαδή το είδος και το πλήθος τιμών που θα διαβιβάζονται στη διαδικασία για επεξεργασία
 - **ο τύπος της επιστρεφόμενης τιμής**, δηλαδή μια δήλωση ενός τύπου δεδομένων της γλώσσας προγραμματισμού Visual Basic

Επιστρεφόμενη Τιμή σε Function

Όταν δηλώνουμε μια συνάρτηση θα πρέπει να υπάρχει οπωσδήποτε μέσα στον κώδικα ορισμού της μια τουλάχιστον εκτελέσιμη εντολή, με την οποία **εκχωρείται στο όνομα της συνάρτησης η τιμή που επιστρέφεται από τη συνάρτηση.**

Ορισμός Διαδικασίας Function

```
εμβέλεια Function ονομα (λίστα παραμέτρων) As τύπος_επιστροφής  
' εντολή – 1  
' εντολή – 2  
...  
' εντολή – κ  
End Sub
```

Εμβέλεια = Public ή Private

Λίστα Παραμέτρων = Μία λίστα με δηλώσεις μεταβλητών
διαχωρισμένες με κόμμα «,»

Για τη λίστα παραμέτρων θα μιλήσουμε σε λίγο πιο αναλυτικά...

Παραδείγματα Functions

```
Function calc(ByVal N1 As Integer, ByVal N2 As Integer) As Integer  
    Return (N1 + N2)      'ή calc = N1+N2  
End Function
```

```
Function calc(ByVal N1 As Integer, ByVal N2 As Integer) As Integer  
    Dim result As Integer  
    result = N1 + N2  
    Return result      'ή calc=result  
End Function
```

```
Function Calc(ByVal N1 As Double, ByVal N2 As Double, ByVal telestis As Char) As Double  
    If telestis = "+" Then Return (N1+N2)  ' ή calc = N1+N2  
    If telestis = "-" Then Return (N1-N2)  ' ή calc = N1-N2  
    If telestis = "*" Then Return (N1*N2)  ' ή calc = N1*N2  
End Sub
```

Η Διαδικασία Sub

- Ορίζεται με δηλωτικές προτάσεις που καθορίζουν την **αρχή** και το **τέλος** της.
- Μεταξύ της αρχής και του τέλους της διαδικασίας τοποθετούνται οι εντολές που θέλουμε να εκτελεστούν
 - οι εντολές μπορεί να είναι εκτελέσιμες προτάσεις που να καλούν άλλες διαδικασίες τύπου Sub ή function καθώς και δηλωτικές προτάσεις
- Στην **αρχική δηλωτική πρόταση** μιας διαδικασίας προσδιορίζονται:
 - η **εμβέλεια (scope) της**, δηλαδή αν θα είναι «ορατή» και θα μπορεί να κληθεί από άλλες ενότητες κώδικα
 - το **όνομά της**, με το οποίο θα αναγνωρίζεται από το σύστημα της γλώσσας προγραμματισμού Visual Basic
 - οι **παράμετροι της**, δηλαδή το είδος και το πλήθος τιμών που θα διαβιβάζονται στη διαδικασία για επεξεργασία

Ορισμός Διαδικασίας Sub

```
εμβέλεια Sub ονομα (λίστα παραμέτρων)  
    ' εντολή – 1  
    ' εντολή – 2  
    ...  
    ' εντολή – κ  
End Sub
```

Εμβέλεια = Public ή Private

Λίστα Παραμέτρων = Μία λίστα με δηλώσεις μεταβλητών
διαχωρισμένες με κόμμα «,»

Για τη λίστα παραμέτρων θα μιλήσουμε σε λίγο πιο αναλυτικά...

Μια απλή διαδικασία Sub

```
Sub simpleCalculator()
```

```
    'Είσοδος δεδομένων
```

```
    Dim Num1, Num2, result As Double, operand As Char
```

```
    Console.WriteLine("Give the 1st number?")
```

```
    Num1 = Console.ReadLine()
```

```
    Console.WriteLine("Give the 2nd number?")
```

```
    Num2 = Console.ReadLine()
```

```
    Console.WriteLine("Give the operand (+,-,*)?")
```

```
    operand = Console.ReadLine()
```

```
    'Επεξεργασία δεδομένων
```

```
    If operand = "+" Then result = Num1 + Num2
```

```
    If operand = "-" Then result = Num1 - Num2
```

```
    If operand = "*" Then result = Num1 * Num2
```

```
    'Εξοδος - Αποτελέσματα
```

```
    Console.WriteLine()
```

```
    Console.WriteLine("*****Program Output*****")
```

```
    Console.WriteLine(Num1 & operand & Num2 & " = " & result)
```

```
End Sub
```

Διαδικασία Sub με παράμετρο

```
Sub sayMessage(ByVal msg As String)  
    Dim stamp As Date  
    stamp = TimeOfDay()  
    MsgBox("Message at " & stamp & msg)  
End Sub
```

Παραδείγματα Διαδικασιών Sub

Sub Main()

'Είσοδος δεδομένων

Dim Num1, Num2, result As Double, operand As Char

Call DataEntry(Num1, Num2, operand)

'Επεξεργασία δεδομένων

Call Calculate(Num1, Num2, operand, result)

'Εξοδος - Αποτελέσματα

Call PrintResults(Num1, Num2, operand, result)

End Sub

Sub DataEntry(ByRef N1 As Double, ByRef N2 As Double, ByRef telestis As Char)

Console.Write("Give the 1st number?")

N1 = Console.ReadLine()

Console.Write("Give the 2nd number?")

N2 = Console.ReadLine()

Console.Write("Give the operand (+,-,*)?")

telestis = Console.ReadLine()

End Sub

Sub Calculate(ByVal N1 As Double, ByVal N2 As Double, ByVal telestis As Char, ByRef apotelesma As Double)

If telestis = "+" Then apotelesma = N1 + N2

If telestis = "-" Then apotelesma = N1 - N2

If telestis = "*" Then apotelesma = N1 * N2

End Sub

Sub PrintResults(ByVal N1 As Double, ByVal N2 As Double, ByVal telestis As Char, ByRef apotelesma As Double)

Console.WriteLine("*****Program Output*****")

Console.WriteLine(N1 & telestis & N2 & " = " & apotelesma)

End Sub

Παράμετροι Διαδικασιών

- Οι παράμετροι σε μια διαδικασία Sub ή Function είναι **υποθετικές μεταβλητές** (λέγονται και **τυπικές παράμετροι - Formal parameters**) που ενδεικτικά προσδιορίζουν τον **τύπο** και το **πλήθος** τιμών οι οποίες θα διαβιβάζονται στη διαδικασία κατά την κλήση της.
- Οι τιμές που διαβιβάζονται στη συνάρτηση **κατά την κλήση της** συνήθως αποκαλούνται **ορίσματα (arguments)** ή **πραγματικές παράμετροι** και είναι παραστάσεις ή μεταβλητές που χρησιμοποιούνται στο τμήμα κώδικα της διαδικασίας.
- Η λίστα παραμέτρων μπορεί να είναι κενή και αν είναι περισσότερες από μια διαχωρίζονται με κόμματα
- Για κάθε παράμετρο αναγράφεται το εικονικό της όνομα και ο τύπος της.

Παράμετροι με Τιμή ή με Αναφορά (ByVal, ByRef)

- Κατά τη δήλωση παραμέτρων σε μια διαδικασία Sub ή Function μπορούμε να καθορίσουμε και το **αν επιτρέπεται ή όχι η τροποποίηση** των τιμών που θα διαβιβαστούν
- Με τη δήλωση **ByVal (παράμετρος με τιμή)** αυτό που διαβιβάζεται δεν επιτρέπεται να τροποποιηθεί (διαβίβαση μόνο της τιμής)
- Με τη δήλωση **ByRef (παράμετρος με αναφορά)** αυτό που διαβιβάζεται επιτρέπεται να τροποποιηθεί (διαβίβαση αναφοράς)

Παραδείγματα ByVal, ByRef με functions

```
**** main program ****
```

```
a = 2 : b = 3
```

```
Sum = calc(a, b)
```

```
Console.WriteLine("Sum= " & Sum & " a= " & a & " b= " & b)
```

```
'-----
```

```
Function calc(ByVal N1 As Integer, ByVal N2 As Integer) As Integer
```

```
    calc = N1+N2
```

```
    N1 = N1+1
```

```
    N2 = N2+1
```

```
End Function
```

```
'-----
```

```
Function calc(ByRef N1 As Integer, ByRef N2 As Integer) As Integer
```

```
    calc = N1+N2
```

```
    N1 = N1+1
```

```
    N2 = N2+1
```

```
End Function
```

Παραδείγματα ByVal, ByRef με Subs

Sub Main()

'Είσοδος δεδομένων

Dim Num1, Num2, result As Double, operand As Char

Call DataEntry(Num1, Num2, operand)

'Επεξεργασία δεδομένων

Call Calculate(Num1, Num2, operand, result)

'Εξοδος - Αποτελέσματα

Call PrintResults(Num1, Num2, operand, result)

End Sub

Sub DataEntry(ByRef N1 As Double, ByRef N2 As Double, ByRef telestis As Char)

Console.Write("Give the 1st number?")

N1 = Console.ReadLine()

Console.Write("Give the 2nd number?")

N2 = Console.ReadLine()

Console.Write("Give the operand (+,-,*)?")

telestis = Console.ReadLine()

End Sub

Sub Calculate(ByVal N1 As Double, ByVal N2 As Double, ByVal telestis As Char, ByRef apotelesma As Double)

If telestis = "+" Then apotelesma = N1 + N2

If telestis = "-" Then apotelesma = N1 - N2

If telestis = "*" Then apotelesma = N1 * N2

End Sub

Sub PrintResults(ByVal N1 As Double, ByVal N2 As Double, ByVal telestis As Char, ByVal apotelesma As Double)

Console.WriteLine("*****Program Output*****")

Console.WriteLine(N1 & telestis & N2 & " = " & apotelesma)

End Sub

Εμβέλεια Διαδικασιών

- Κατά τη δήλωση μιας διαδικασίας μπορούμε να χρησιμοποιήσουμε τις εντολές **Private & Public** :
 - Με τη δήλωση **Public** δηλώνουμε διαδικασίες που θέλουμε να έχουν **καθολική εμβέλεια**, δηλαδή να είναι ορατές από όλα τα σημεία-ενότητες κώδικα του προγράμματός μας
 - Με τη δήλωση **Private** δηλώνουμε διαδικασίες που θέλουμε να έχουν **τοπική εμβέλεια** δηλαδή μόνο μέσα στην ενότητα κώδικα που ανήκουν

Παράδειγμα Εμβέλειας Διαδικασιών

Module Mod_1

Public Sub Mod_1_Sub_1

Private Sub Mod_1_Sub_2

Public Function Mod_1_F1

Private Function Mod_1_F2

Module Mod_2

Public Sub Mod_2_Sub_1

Private Sub Mod_2_Sub_2

Public Function Mod_2_F1

Private Function Mod_2_F2

«Ορατές» διαδικασίες στην ενότητα Mod_2 από την ενότητα Mod_1 είναι οι Mod_1_Sub_1 και Mod_1_F1 ενώ στην ενότητα Mod_1 από την ενότητα Mod_2 είναι οι διαδικασίες Mod_2_Sub_1 και Mod_2_F1

Εμβέλεια Διαδικασιών & Μεταβλητών

```
Module Mod_1  
  Public A As Integer  
  Private B As Integer
```

```
Public Sub Mod_1_Sub_1  
  Dim C As Integer
```

```
Private Function Mod_1_F1  
  Dim A As Integer  
  Dim C As Integer
```

«Ορατές» διαδικασίες έξω από την ενότητα Mod_1 είναι μόνο η Mod_1_Sub_1 και «ορατές» μεταβλητές μόνο η A.

Μέσα στην ενότητα Mod_1 «ορατές» είναι τόσο οι διαδικασίες Mod_1_Sub_1 και Mod_1_F1 όσο και οι μεταβλητές A, B. ΠΡΟΣΟΧΗ στην Mod_1_F1 ισχύει η τοπική μεταβλητή A

Στατικές Μεταβλητές

```
Module Mod_1  
  Public A As Integer  
  Private B As Integer
```

```
  Public Sub Sub_1  
    Static D As Integer
```

Η μεταβλητή D διατηρεί την τιμή της και μετά την ολοκλήρωση της διαδικασίας Sub_1

Τέλος Ενότητας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

