



ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα **ΠΠ**



Εισαγωγή στους Υπολογιστές

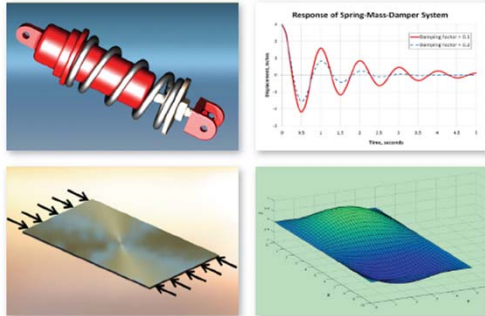
Ενότητα #5: Διαγράμματα ροής (Flow Charts), Δομές επανάληψης

Καθ. Δημήτρης Ματαράς
Πολυτεχνική Σχολή
Τμήμα Χημικών Μηχανικών

Διαγράμματα ροής (Flow Charts), Δομές επανάληψης

BASIC ENGINEERING SERIES AND TOOLS

ENGINEERING COMPUTATIONS
AN INTRODUCTION
USING MATLAB® AND EXCEL®



JOSEPH C. MUSTO • WILLIAM E. HOWARD • RICHARD R. WILLIAMS

MATLAB Programming

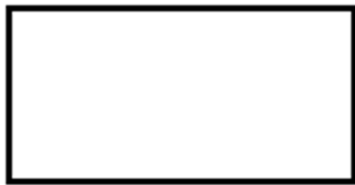
A. Καλαμπούνιας

Διαγράμματα ροής

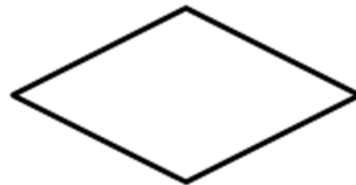
- Τα **διαγράμματα ροής** είναι διαγράμματα που απεικονίζουν την πορεία που ακολουθείται στους υπολογισμούς
- Είναι βασικό εργαλείο στο **σχεδιασμό** πολύπλοκων **αλγορίθμων**
- Είναι επίσης πολύ χρήσιμα για την **επεξήγηση** των **αλγορίθμων**, ακόμα και των πιο απλών

Από τι μπορεί να αποτελείται ένα διάγραμμα ροής

- Παρουσιάζονται τα **βασικότερα μέρη** που μπορεί να εμφανιστούν σε ένα διάγραμμα ροής



Program Step



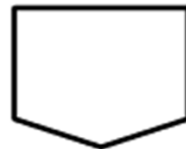
Decision Point



Input/Output



Connector



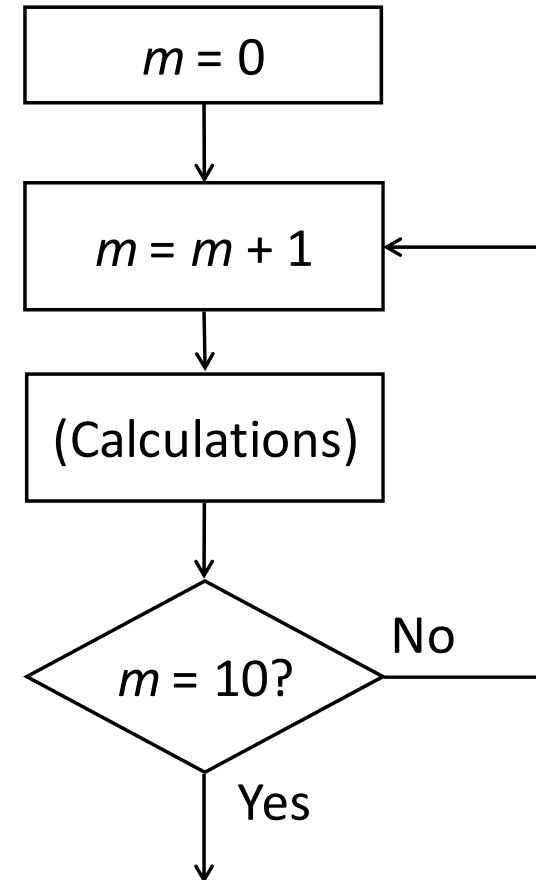
Off-Page
Connector



Start/Finish

Η δομή *for* στο MATLAB

- Ονομάζεται επίσης δομή *do* σε άλλες γλώσσες προγραμματισμού
- Χρησιμοποιείται όταν θέλουμε να **επαναλάβουμε μια διαδικασία πολλές φορές**
- Στο παράδειγμα, οι υπολογισμοί επαναλαμβάνονται 10 φορές



Η δομή *for* στο MATLAB

- Στο **MATLAB**, η δομή *for* ξεκινά δηλώνοντας πόσες φορές θα γίνει η επανάληψη
- Χρησιμοποιείται ένας **μετρητής** για αυτή τη δήλωση
- Παραδείγματα:

```
for k = 1:100
```

(counter = k , η επανάληψη θα επαναληφθεί 100 φορές)

```
for i = 1:2:7
```

(counter = i , ο μετρητής θα μεταβάλλεται με βήμα 2 κάθε φορά μέχρι να φθάσει την τιμή 7. Άρα η δομή θα επαναληφθεί 4 φορές ($i = 1, 3, 5$ και 7))

Η δομή *for* στο MATLAB

- Η δομή **τελειώνει** με την εντολή ***end***
- Στα **M-files**, ο **MATLAB editor** αυτόματα θα εισάγει το *for* και το *end*, οπότε ο χρήστης θα προσθέσει το υπόλοιπο κείμενο:

```
1   for j = 1:10
2       x(j) = 5*j;
3   end
```

Παράδειγμα του *for* Loop

```
1   for j = 1:10
2       x(j) = 5*j;
3   end
```

- Η δομή ξεκινά με τιμή, $j = 1$
- Λόγω των παρενθέσεων καταλαβαίνουμε ότι το x θα είναι μονοδιάστατος πίνακας
- Το $x(1)$ θα είναι ίσο με $5*1 = 5$
- Στη δεύτερη επανάληψη, $j = 2$
- Το $x(2)$ θα είναι ίσο με $5*2 = 10$
- Αυτό θα επαναληφθεί μέχρι $j = 10$ και τελικά θα έχουμε $x(10) = 50$

Παράδειγμα του *for* Loop

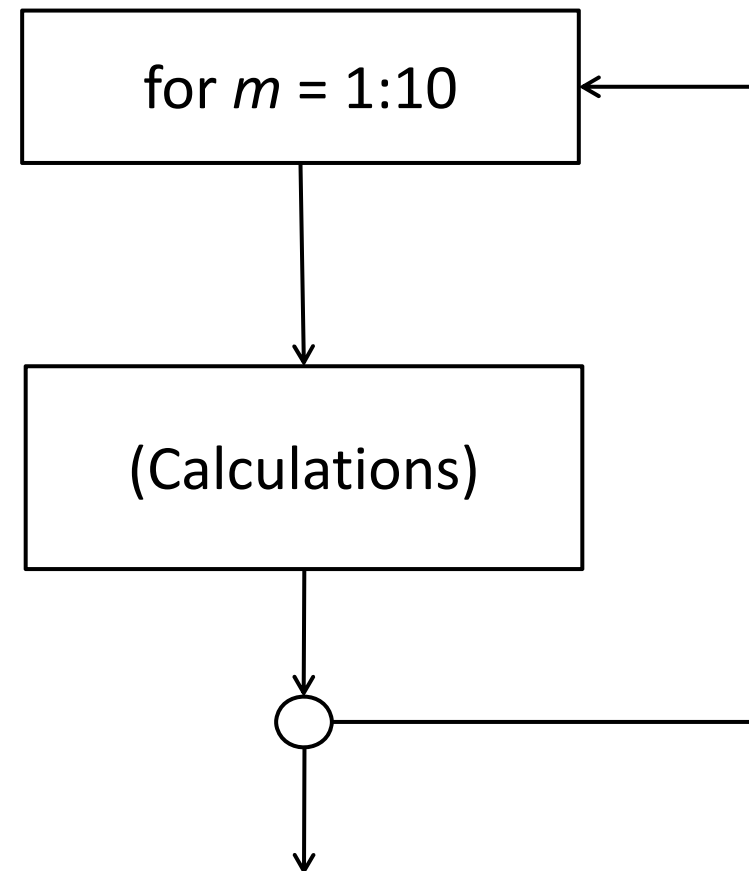
```
1   for j = 1:10
2       x(j) = 5*j;
3   end
```

- Το x θα είναι ένας μονοδιάστατος πίνακας με 10 στοιχεία:

```
>> x
x =
    5    10    15    20    25    30    35    40    45    50
```

Γενική μορφή του *for* σε ένα διάγραμμα ροής

- Χρησιμοποιούνται βέλη (connector) που δείχνουν την πορεία εκτέλεσης του αλγορίθμου



Η δομή *for* Loop στην τρέχουσα οθόνη (Interactive Mode)

- Οι εντολές των δομών μπαίνουν **απευθείας** στο **command prompt**
- Οι υπολογισμοί δεν εκτελούνται μέχρι να εισαχθεί η δήλωση `end`

```
>> for j = 1:10
x(j) = j*5;
end
>> x
x =
    5    10    15    20    25    30    35    40    45    50
```

Η δομή *for* Loop στην τρέχουσα οθόνη (Interactive Mode)

- Αν παραληφθεί το ερωτηματικό, τότε τα αποτελέσματα θα εμφανίζονται στην οθόνη σε κάθε επανάληψη:

```
>> clear x
>> for j = 1:10
x(j) = j*5
end
x =
    5
x =
    5    10
x =
    5    10    15
x =
    5    10    15    20
x =
    5    10    15    20    25
x =
    5    10    15    20    25    30
x =
    5    10    15    20    25    30    35
x =
    5    10    15    20    25    30    35    40
x =
    5    10    15    20    25    30    35    40    45
x =
    5    10    15    20    25    30    35    40    45    50
```

Παραδείγματα *for* Loop

```
y = 0;
```

```
for k = 1:5           (1, 2, 3, 4, 5)
```

```
    y = y + k;       ((( (0+1) +2) +3) +4) +5
```

```
end
```

```
y
```

y =
15

Παραδείγματα *for* Loop

```
y = 0;  
for k = 2:2:8  
    y = y + k;  
end  
y
```

y =
20

Παραδείγματα *for* Loop

```
for k = 1:5
    y(k) = k^2;
end
y
```

```
y =
    1     4     9    16    25
```

Παραδείγματα *for* Loop

```
for j = 1:3
    for k = 1:3
        T(j,k) = j*k;
    end
end
```

end

T

```
T =
     1     2     3
     2     4     6
     3     6     9
```


Παραδείγματα *for* Loop

- Δίνεται η εξίσωση:

$$y = 2^{0.4x} + 5$$

- Θέλουμε να κάνουμε γραφική παράσταση της εξίσωσης για x από -10 μέχρι 10
- Θα χρησιμοποιηθεί η δομή *for* για τον υπολογισμό και αποθήκευση των τιμών x και y σε πίνακες μιας διάστασης

Παραδείγματα *for* Loop

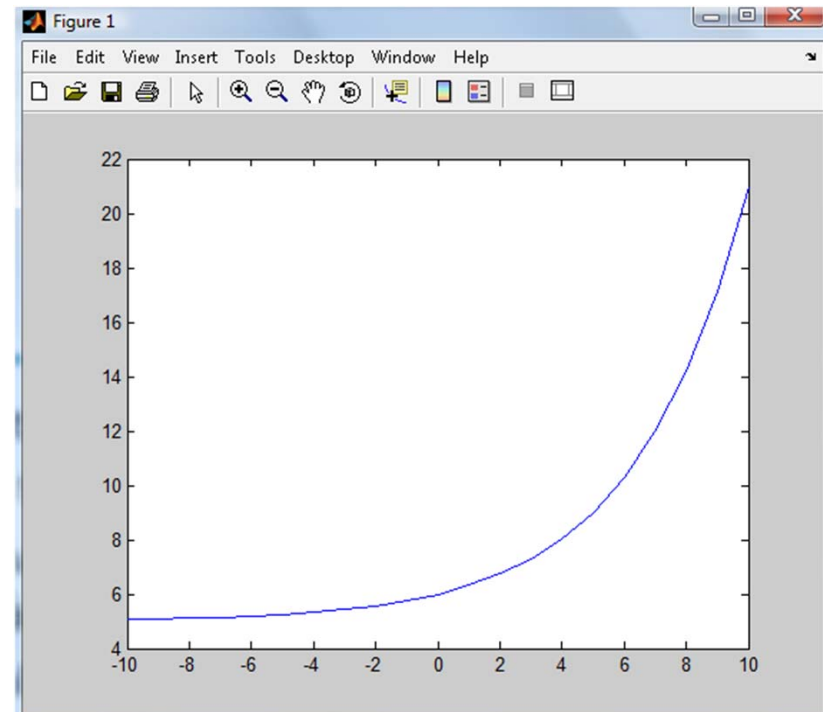
$$y = 2^{0.4x} + 5$$

```
for i = 1:21
    x(i) = -10 + (i-1);
    y(i) = 2^(0.4*x(i)) + 5;
end
```

- Μετά την εκτέλεση του κώδικα, δημιουργούνται δύο μονοδιάστατοι πίνακες, x και y , με 21 στοιχεία

Η εντολή **Plot**

- Ο αποθηκευμένοι πίνακες μπορούν να **αναπαρασταθούν γραφικά** με την εντολή : **plot(x,y)**
- Οποιοιδήποτε δύο μονοδιάστατοι πίνακες μπορούν να αναπαρασταθούν, αρκεί να έχουν το **ίδιο μέγεθος**
- Το γράφημα θα εμφανιστεί σε **νέο παράθυρο**

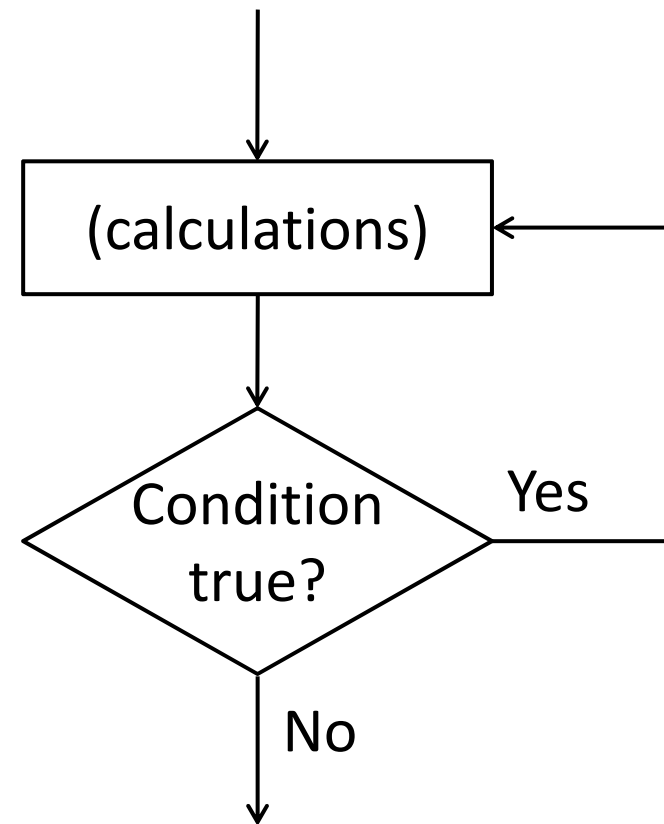


Οι δομές *for* και *while* στο MATLAB

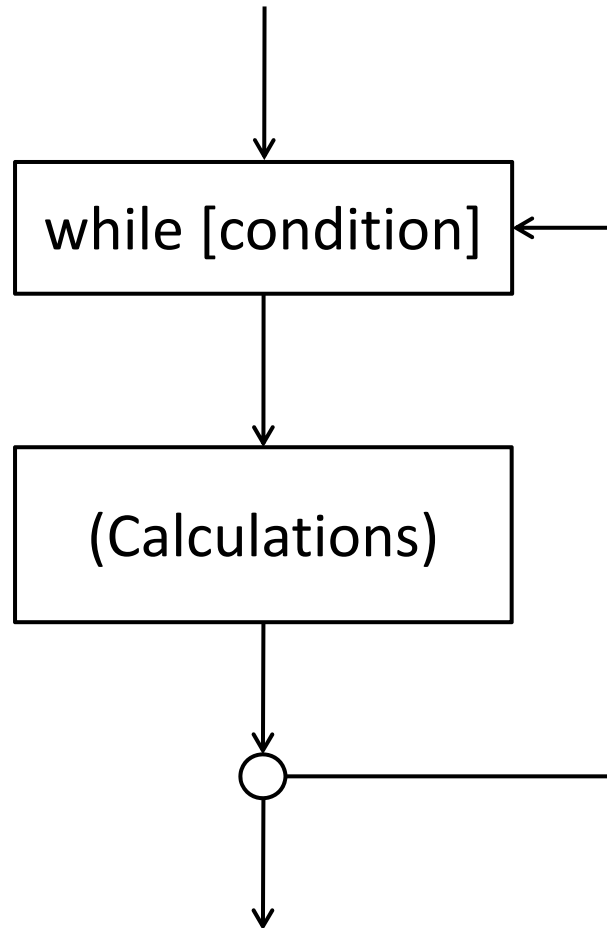
- Η δομή *for* θα εκτελεστεί για ένα συγκεκριμένο αριθμό επαναλήψεων
- Η δομή *while* θα συνεχίσει να επαναλαμβάνεται μέχρι να ικανοποιηθεί κάποια συνθήκη
- **Παραδείγματα:** Θεωρούμε ένα χρηματικό ποσό που θέλουμε να επενδυθεί με δεδομένο τόκο
- Εάν θέλουμε να υπολογίσουμε **το ύψος του ποσού μετά από 10 χρόνια**, τότε χρησιμοποιούμε τη δομή *for*
- Εάν θέλουμε να προσδιορίσουμε πόσος χρόνος χρειάζεται **μέχρι το ποσό να φθάσει το ύψος των \$100.000**, τότε πρέπει να χρησιμοποιήσουμε τη δομή *while*

Η δομή *while* σε διάγραμμα ροής

- Η πρώτη γραμμή θα είναι:
`while (condition)`
- Η τελευταία γραμμή θα είναι:
`end`



Η γενική μορφή της δομής *while* σε ένα διάγραμμα ροής



Η γενική μορφή της δομής *while* συνδιασμένη με **and, or**

- Έλεγχος δύο συνθηκών ταυτόχρονα θα είναι:
`while a<100 & b>=50;`
- Έλεγχος μιας συνθήκης ή εναλλακτικά μιας άλλης συνθήκης θα είναι:
`while a<100 | b>=50;`

Η εισαγωγή του συμβόλου | γίνεται με shift+\ από το πληκτρολόγιο.

- Η τελευταία γραμμή στις δομές αυτές πάντα θα είναι:
`end`

Παράδειγμα

- Θεωρούμε τη δομή :

```
k = 0;
```

```
while k < 10
```

```
    k = k + 2
```

```
end
```

- Πόσες φορές θα εκτελεστεί η δομή?

Αρχικά, $k = 0$, άρα έχουμε είσοδο στη δομή

Επανάληψη #1: $k = 2$, άρα η εκτέλεση συνεχίζεται

Επανάληψη #2: $k = 4$, άρα η εκτέλεση συνεχίζεται

Επανάληψη #3: $k = 6$, άρα η εκτέλεση συνεχίζεται

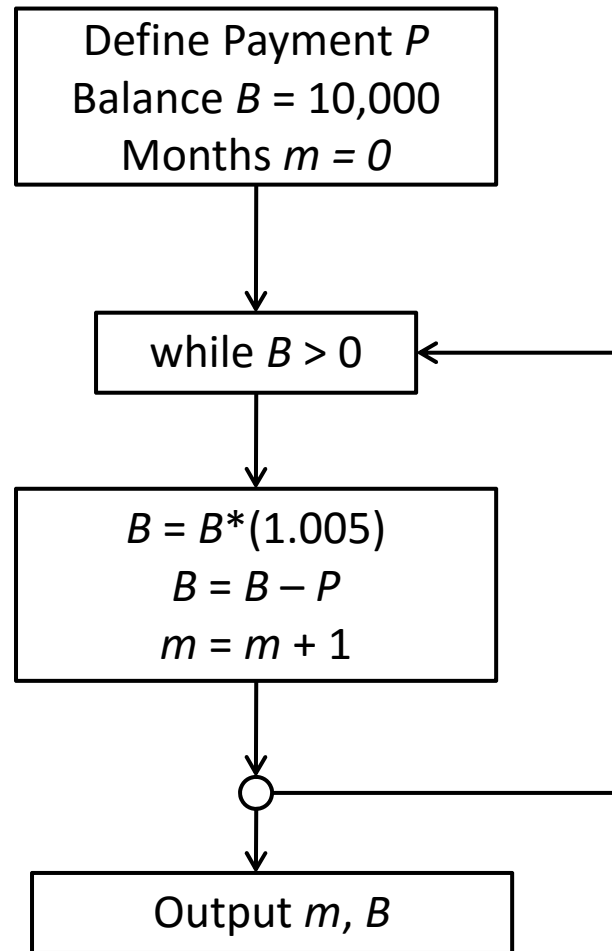
Επανάληψη #4: $k = 8$, άρα η εκτέλεση συνεχίζεται

Επανάληψη #5, $k = 10$, άρα το k δεν είναι λιγότερο από 10 και η εκτέλεση τερματίζεται

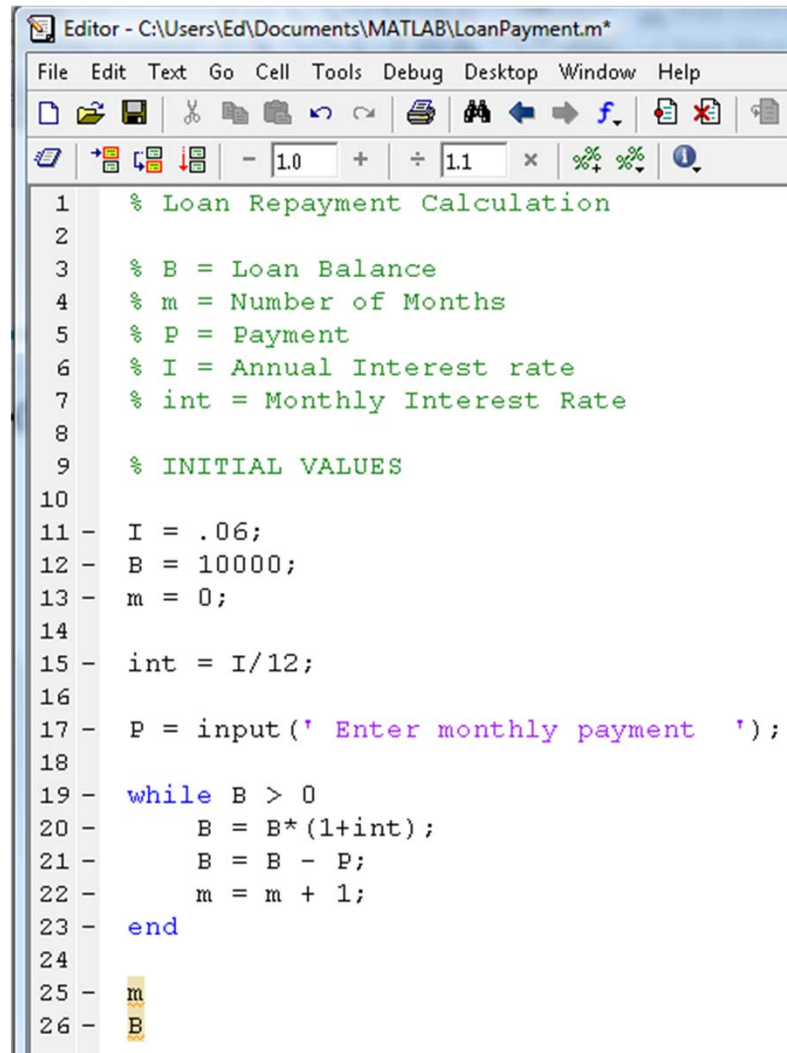
Παράδειγμα του *while*

- Λαμβάνουμε δάνειο \$10,000 με μηνιαίο επιτόκιο 6%.
- Άρα, κάθε μήνα, το ποσό που χρωστάμε αυξάνεται κατά 0.5% (6% δια του 12) και μειώνεται με την πληρωμή της μηνιαίας δόσης στην τράπεζα από εμάς
- Πόσους μήνες θα χρειαστούν μέχρι την ολική εξόφληση του δανείου, όταν η μηνιαία πληρωμή θα είναι:
 - a. \$500
 - b. \$200
 - c. \$100

Διάγραμμα ροής της επίλυσης



Το **m-File** και η εντολή **input**



```
Editor - C:\Users\Ed\Documents\MATLAB\LoanPayment.m*
File Edit Text Go Cell Tools Debug Desktop Window Help
1 % Loan Repayment Calculation
2
3 % B = Loan Balance
4 % m = Number of Months
5 % P = Payment
6 % I = Annual Interest rate
7 % int = Monthly Interest Rate
8
9 % INITIAL VALUES
10
11 - I = .06;
12 - B = 10000;
13 - m = 0;
14
15 - int = I/12;
16
17 - P = input(' Enter monthly payment ');
18
19 - while B > 0
20 -     B = B*(1+int);
21 -     B = B - P;
22 -     m = m + 1;
23 - end
24
25 - m
26 - B
```

- Προσοχή στη χρήση της εντολής **input**: η τιμή που θα εισαχθεί στο prompt θα ανατεθεί στη μεταβλητή **P**.
- Το **μήνυμα** εισάγεται μέσα σε **μονά εισαγωγικά!**

Αποτελέσματα

- πληρωμή= \$500:

```
>> LoanPayment
Enter monthly payment 500
m =
    22
B =
-437.4940
```

- Το δάνειο θα εξοφληθεί σε 22 μήνες
- Μετά από 22 μηνιαίες πληρωμές, η διαφορά των \$437 θα επιστραφεί στον δανειολήπτη

Αποτελέσματα

- Πληρωμή = \$200

```
>> LoanPayment
Enter monthly payment 200
m =
    58
B =
 -63.8643
```

- Πληρωμή = \$100

```
>> LoanPayment
Enter monthly payment 100
m =
   139
B =
  -2.4219
```

Αποτελέσματα

- Δοκιμάζοντας πληρωμή = \$45:

```
>> LoanPayment  
Enter monthly payment 45
```

- Οι υπολογισμοί θα συνεχίζονται μέχρι να πατήσουμε ctrl+C για να τερματιστεί η εκτέλεση

```
>> LoanPayment  
Enter monthly payment 45  
Error in ==> LoanPayment at 23  
end
```

Αποτελέσματα

- Ελέγχοντας τις τιμές m και B μετά το τέλος των υπολογισμών:

```
>> m
m =
    18999932
>> B
B =
    Inf
```

- Αν ο χρόνος αποπληρωμής είναι μεγαλύτερος από 1.000.000 χρόνια, θα χρωστάμε περισσότερα χρήματα από αυτά που μπορεί αν υπολογίσει το MATLAB.

Αέναιεε επαναλήψει (Infinite Loops)

- Όταν χρησιμοποιούμε δομή **while**, υπάρχει ο κίνδυνος να μην τερματίζεται ποτέ η επανάληψη
- Καθώς ο τερματισμός της επανάληψης εξαρτάται από το αν πληρείται μια συνθήκη (σε αυτή την περίπτωση, η συνθήκη $< ή = 0$), είναι πιθανό η συνθήκη να μην πληρείται ποτέ και η επανάληψη να μην τερματίζεται ποτέ

παράδειγμα

- Ποιές είναι οι τιμές του A και του m μετά την εκτέλεση των εντολών:

```
m = 0;  
A = 20;  
while A <= 50  
    A = A + 5;  
    m = m + 1;  
end  
A  
m
```

```
A =  
    55  
m =  
    7
```

παράδειγμα

- Ποιές είναι οι τιμές του A και του m μετά την εκτέλεση των εντολών:

```
m = 0;
```

```
A = 100;
```

```
while A > 15
```

```
    A = A/2;
```

```
    m = m + 1;
```

```
end
```

```
A
```

```
m
```

```
A =  
    12.5000  
m =  
     3
```

παράδειγμα

- Ποιές είναι οι τιμές του A και του m μετά την εκτέλεση των εντολών:

```
m = 0;
```

```
A = 10;
```

```
while A > 0
```

```
    A = sqrt(A);
```

```
    m = m + 1;
```

```
end
```

```
A
```

```
m
```

Αέναη επανάληψη: η τετραγωνική ρίζα δεν θα γίνει ποτέ μηδέν

Αποθήκευση σε αρχείο

- Όταν πρόκειται να προκύψει μεγάλος αριθμός δεδομένων, είναι χρήσιμο να **εγγραφούν τα δεδομένα απ'ευθείας σε κάποιο αρχείο**.
- Αρχικά, πρέπει να οριστεί το όνομα του αρχείου. Αυτό γίνεται με την εντολή `fopen`:

`Fid=fopen('filename', 'attribute')`

Το όνομα του αρχείου αντιστοιχείται σε μια μεταβλητή με όνομα της επιλογής μας (π.χ. με έκταση **.txt** όταν πρόκειται για αρχείο μορφής **text**). Το αρχείο σε αυτή την περίπτωση θα ανοίγει με το Notepad των Windows.

Αποθήκευση σε αρχείο

- Το **attribute** θα έχει δύο δυνατές τιμές, το **'wt'** και το **'at'**.
- Το **'wt'** θα χρησιμοποιείται όταν έχουμε **αρχείο τύπου text**
- Όταν δεν υπάρχει ήδη το αρχείο, θα δημιουργείται.
- Όταν υπάρχει ήδη το αρχείο, τότε **το περιεχόμενο του θα ανανεώνεται γράφοντας πάνω στα παλιά** τα νέα δεδομένα (θα γίνεται δηλαδή **overwrite** των νέων τιμών πάνω στις παλιές τιμές εντός του αρχείου).
- Το **'at'** θα χρησιμοποιείται όταν έχουμε αρχείο, το οποίο **προϋπάρχει** και θέλουμε τα **νέα δεδομένα να προστεθούν στα παλιά, χωρίς να τα σβήσουν** (append data)
- Το αρχείο που κατασκευάζεται βρίσκεται στο **current directory**

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Πατρών**» 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.



Σημείωμα Αναφοράς

Copyright©2014 Πανεπιστήμιο Πατρών, Τμήμα Χημικών Μηχανικών,
Καθ. Δημήτρης Ματαράς (mataras@upatras.gr), «Εργαστήριο
Υπολογιστών» «Διαγράμματα ροής (Flow Charts), Δομές επανάληψης»

Διαθέσιμο από τη δικτυακή διεύθυνση:

<https://eclass.upatras.gr/courses/CMNG2112/>

Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.