

ΑΛΓΟΡΙΘΜΟΙ

Δρ. Χάρης Κουζινόπουλος

Εθνικό Κέντρο Έρευνας και Τεχνολογικής Ανάπτυξης

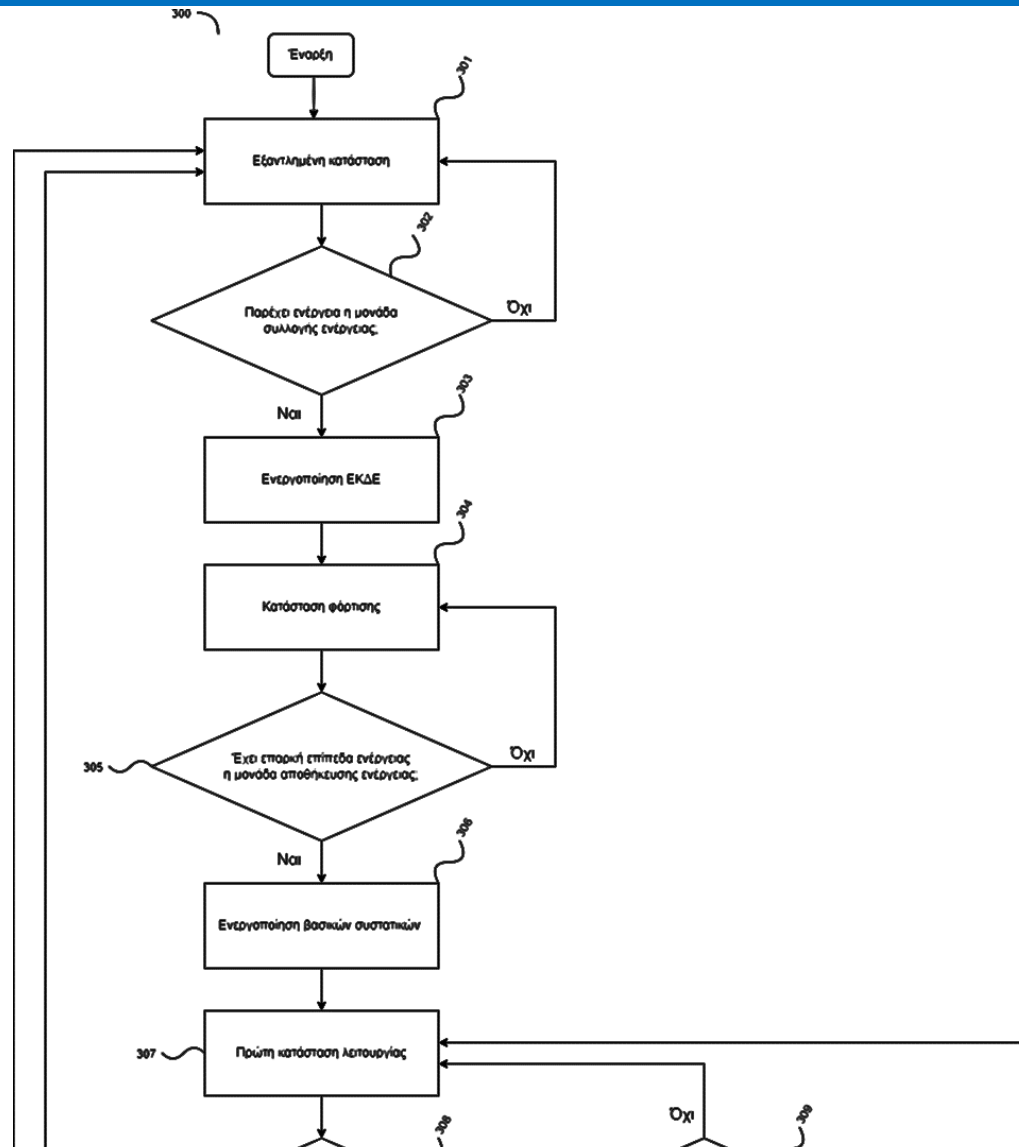
Πανεπιστήμιο Μακεδονίας

Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης

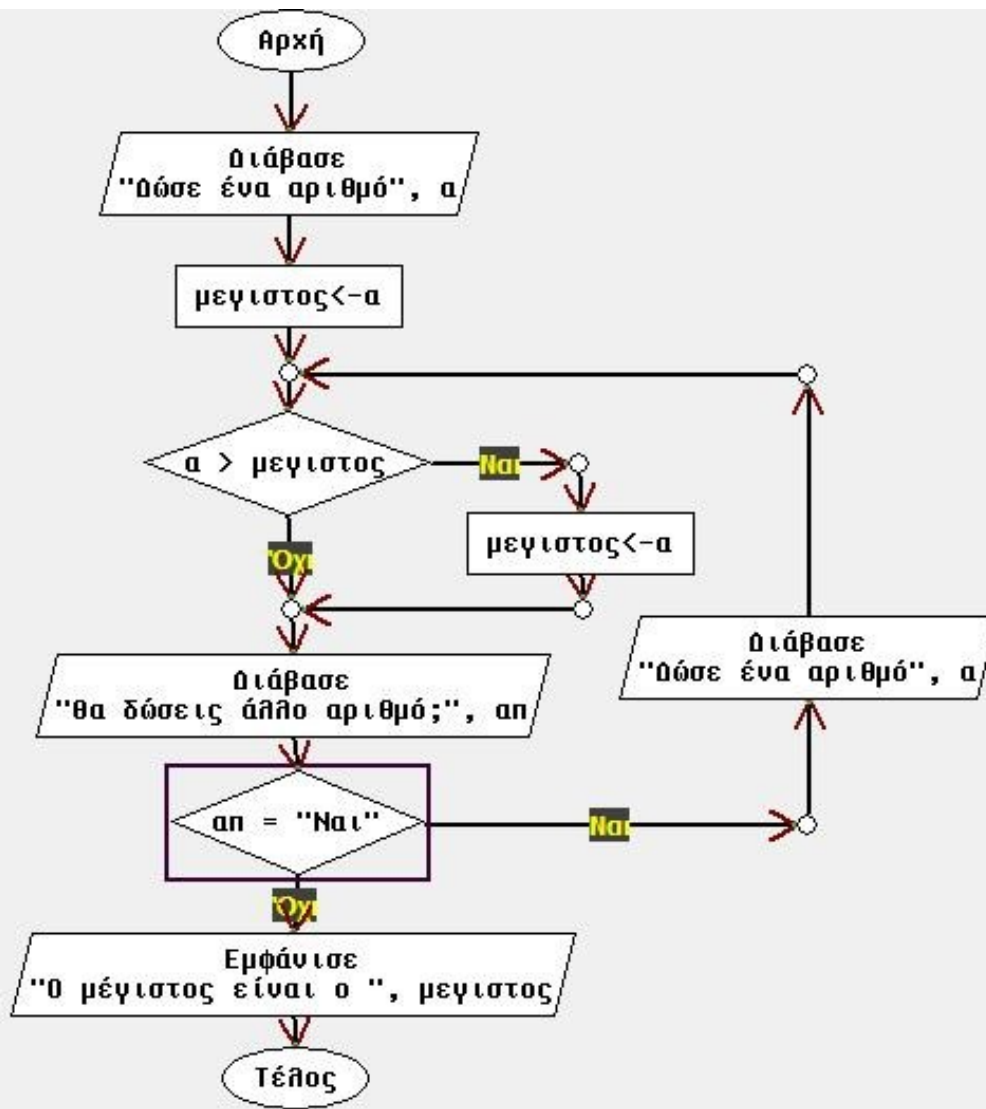


CMOR
Computational Methodologies
& Operations Research

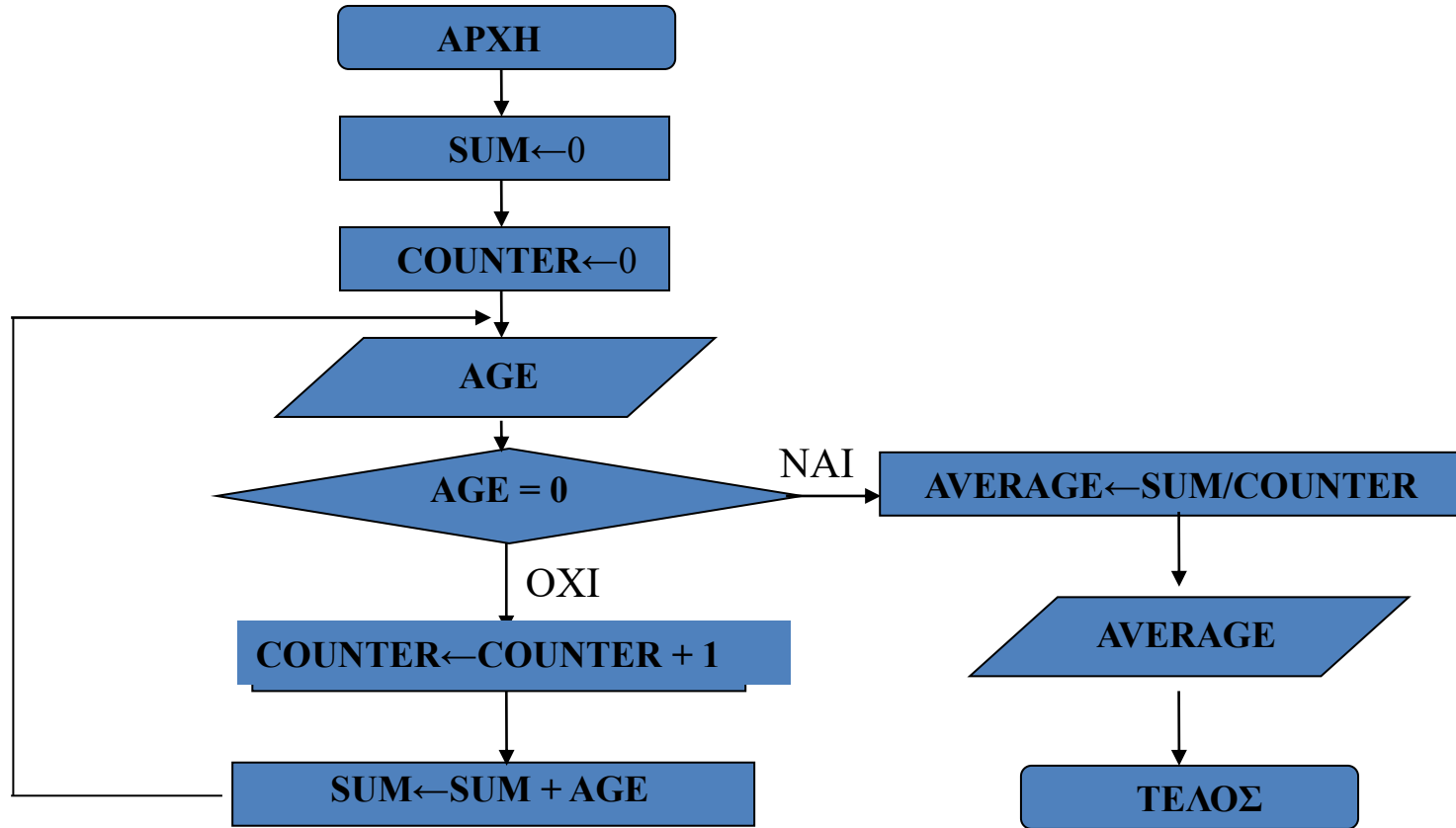
Τρόποι αναπαράστασης Αλγορίθμων



Τρόποι αναπαράστασης Αλγορίθμων



Τρόποι αναπαράστασης Αλγορίθμων



Ποια είναι η λειτουργία του?

Περιγραφή Αλγορίθμων

Χρήση Ψευδοκώδικα – Γενική μορφή ψευδοκώδικα

<i>Αλγόριθμος</i> : όνομα αλγορίθμου	
<i>Είσοδος</i> : ονόματα μεταβλητών	
<i>Έξοδος</i> : ονόματα μεταβλητών	
1	εντολή_1
2	εντολή_2
...	...
n	εντολή_n

Τελεστές & Συνθήκες – [1]

Οι εντολές ψευδοκώδικα περιέχουν

- Μαθηματικές παραστάσεις
- Συνθήκες
 - Σχεσιακούς τελεστές $<$, \leq , $>$, \geq , $=$
π.χ. $(x \geq y)$, $(z < y)$
 - Λογικοί τελεστές: και, ή, ή, όχι,
π.χ. $(x \geq y)$ και $(y < z)$, $(x \geq y)$ ή $(y < z)$
π.χ. $(x \geq y)$ ή $(y < z)$

Προτασιακή λογική

**Ο τομέας της λογικής
που ασχολείται με προτάσεις**

Σύζευξη (conjunction)

- " p και q "

p	q	$p \wedge q$
F	F	F
F	T	F
T	F	F
T	T	T

Παράδειγμα

«Σήμερα είναι Παρασκευή.»

«Σήμερα δεν βρέχει.»

Σύζευξη:

«Σήμερα είναι Παρασκευή και δεν βρέχει.»

Διάζευξη (disjunction)

- " $p \text{ ή } q$ "

p	q	$p \vee q$
F	F	F
F	T	T
T	F	T
T	T	T

Παράδειγμα

«Σήμερα θα πάμε στο σινεμά.»

«Σήμερα θα πάμε στο θέατρο.»

Διάζευξη:

**«Σήμερα θα πάμε στο σινεμά ή θα πάμε στο
θέατρο.»**

Αποκλειστική Διάζευξη (disjunction)

- " p ή q αλλά όχι και τα δύο"

p	q	$p \oplus q$
F	F	F
F	T	T
T	F	T
T	T	F

Παράδειγμα

«Ένας φυσικός αριθμός είναι άρτιος.»

«Ένας φυσικός αριθμός είναι περιττός.»

Αποκλειστική Διάζευξη:

«Ένας φυσικός αριθμός είτε είναι άρτιος είτε είναι περιττός.»

(δεν μπορεί να είναι και τα δύο)

Λογικές Πράξεις: Άρνηση (negation)

- "όχι" p :

Πίνακας Αληθείας

p	$\neg p$
F	T
T	F

Παράδειγμα

«Σήμερα είναι Παρασκευή»

«Ένας φυσικός αριθμός είναι άρτιος.»

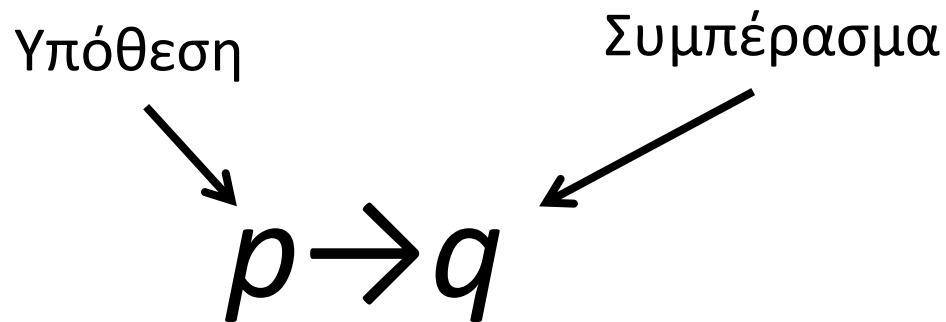
Άρνηση:

«Σήμερα δεν είναι Παρασκευή»

«Ένας φυσικός αριθμός δεν είναι άρτιος.»

Συνεπαγωγή (implication)

- « p συνεπάγεται την q »
- «Αν p τότε q »



Η συνεπαγωγή διαφέρει από τις υπόλοιπες πράξεις λόγω της αιτιότητας που εγγενώς έχει. Εμείς δεν θα ασχοληθούμε με αυτό μιας και είναι περισσότερο θέμα φιλοσοφίας. Μας ενδιαφέρει μόνο ο πίνακας αληθείας της.

Συνεπαγωγή (implication)

- « p συνεπάγεται την q »
- «Αν p τότε q »

Ο πίνακας αληθείας της συνεπαγωγής είναι ψευδής **μόνο** όταν $p = T$ και $q = F$. Γιατί;

	p	q	$p \rightarrow q$	Υπόθεση	Συμπέρασμα
	T	T	T		
	T	F	F		
	F	T	T		
	F	F	T		

Συνεπαγωγή (implication)

- « p συνεπάγεται την q »
- «Αν p τότε q »
- Έστω η συνεπαγωγή:
 - «Εάν έχει ήλιο, θα φορέσω αντηλιακό»
- Πότε δεν θα ικανοποιήσω την υπόσχεσή μου;
 - Εάν έχει ήλιο και φορέσω αντηλιακό;
 - Εάν δεν έχει ήλιο;
 - Εάν έχει ήλιο και δεν φορέσω αντηλιακό;

Συνεπαγωγή (implication)

$$p \rightarrow q \equiv (\neg p \vee q)$$

p	q	$\neg p$	$p \rightarrow q$	$\neg p \vee q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Τελεστές & Συνθήκες – [3]

ΠΑΡΑΔΕΙΓΜΑ.

$F = (A \text{ AND } (\text{NOT } B)) \text{ OR } ((\text{NOT } A) \text{ AND } B \text{ AND } C)$

Τελεστές & Συνθήκες – [3]

ΠΑΡΑΔΕΙΓΜΑ.

$$F = (A \text{ AND } (\text{NOT } B)) \text{ OR } ((\text{NOT } A) \text{ AND } B \text{ AND } C)$$

A	B	C	!A	!B	A!B	!ABC	F
0	0	0	1	1	0	0	0
0	0	1	1	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	1	1
1	0	0	0	1	1	0	1
1	0	1	0	1	1	0	1
1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0

Ψευδοκώδικας – [1]

Ο ψευδοκώδικας (pseudocode) αποτελεί συνδυασμό φυσικής γλώσσας και δομών προγραμματισμού υψηλού επιπέδου.

Δεν υπάρχει ακριβής ορισμός της γλώσσας ψευδοκώδικα, ούτε υπάρχει μια αυστηρά καθορισμένη μορφή της.

Οι μορφές του ψευδοκώδικα που επιλέγονται θυμίζουν σύγχρονες γλώσσες προγραμματισμού υψηλού επιπέδου, όπως είναι η Python, η C/C++ και η Java.

Algorithm 2 The Bellman-Kalaba algorithm

```
1: procedure BELLMANKALABA( $G, u, l, p$ )
2:   for all  $v \in V(G)$  do
3:      $l(v) \leftarrow \infty$ 
4:   end for
5:    $l(u) \leftarrow 0$ 
6:   repeat
7:     for  $i \leftarrow 1, n$  do
8:        $min \leftarrow l(v_i)$ 
9:       for  $j \leftarrow 1, n$  do
10:        if  $min > e(v_i, v_j) + l(v_j)$  then
11:           $min \leftarrow e(v_i, v_j) + l(v_j)$ 
12:           $p(i) \leftarrow v_j$ 
13:        end if
14:      end for
15:       $l'(i) \leftarrow min$ 
16:    end for
17:    changed  $\leftarrow l \neq l'$ 
18:     $l \leftarrow l'$ 
19:  until not changed
20: end procedure
```

```
[BEGIN C-style pseudo code]

n = 0; // the number of  $P_K$  found
for(c = 1; c <= 8; c++){
  if( L( $x_c$ ) == 1 ){
    D = c; // the direction code of  $P_1^*$  from '*'
     $P_1^* = x_c$ ;
    FindP( $P_1^*, D, 1$ );
  }
}

FindP( $x_0, D, k$ ){
  k++; // increase the distance of  $x_0$  from '*'
  for(c = 1; c <= 8; c++){
    if( L( $x_c$ ) == k &&
       (c == D || c == D+1 || c == D-1) ){
      if(k < K){
         $P_k^* = x_c$ ;
        FindP( $P_k^*, D, k$ ); // recursion
      }
      else if(k == K){
        n++; // the number of  $P_K$  found
         $P_K^n = x_c$ ; // Add  $P_k^*$  to  $\{P_K^n\}_{k=1}^n$ 
      }
    }
  }
}

[END C-style pseudo code]
```

Ψευδοκώδικας – [2]

Σύνολο βασικών πράξεων οι οποίες χρησιμοποιούνται σε ψευδοκώδικα:

- Εκχώρηση τιμής σε μεταβλητή
- Εκτέλεση αριθμητικών πράξεων
- Σύγκριση δυο αριθμών
- Κλήση συναρτήσεων
- Κλήση με τιμή
- Κλήση με αναφορά
- Χρήση δεικτών

Ψευδοκώδικας – [3]

Απλή εντολή καταχώρησης

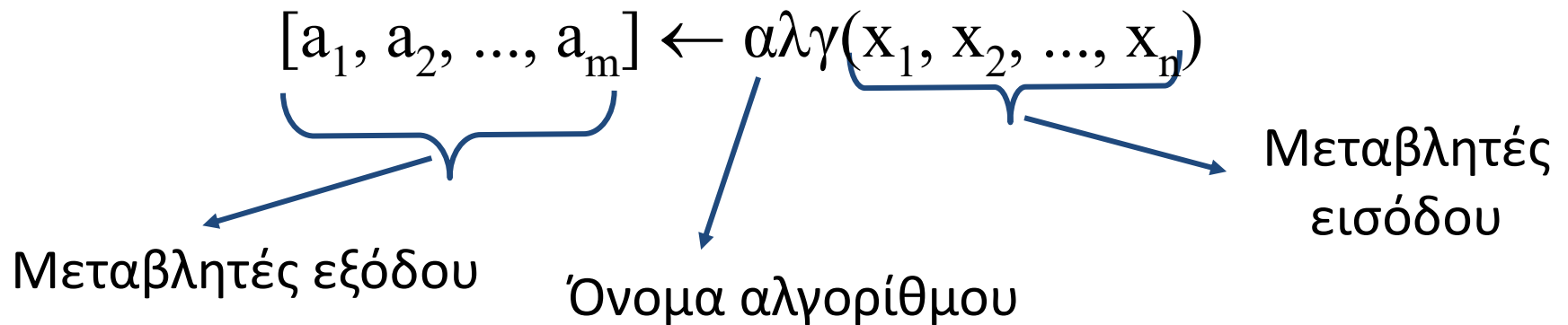
όνομα_μεταβλητής \leftarrow μαθηματική έκφραση,
$$z \leftarrow z + 2x + 5 + y^2$$

Σύνθετη εντολή καταχώρησης

όνομα_μεταβλητής \leftarrow Συστοιχία ή Πίνακας ή παραστάσεις
πινάκων
$$z \leftarrow [2 \ 10 \ 3 \ 8 \ -4 \ 2 \ -8]$$

Ψευδοκώδικας – [4]

Κλήση αλγορίθμων : Στην εντολή καταχώρησης μπορούμε να χρησιμοποιούμε αλγορίθμους

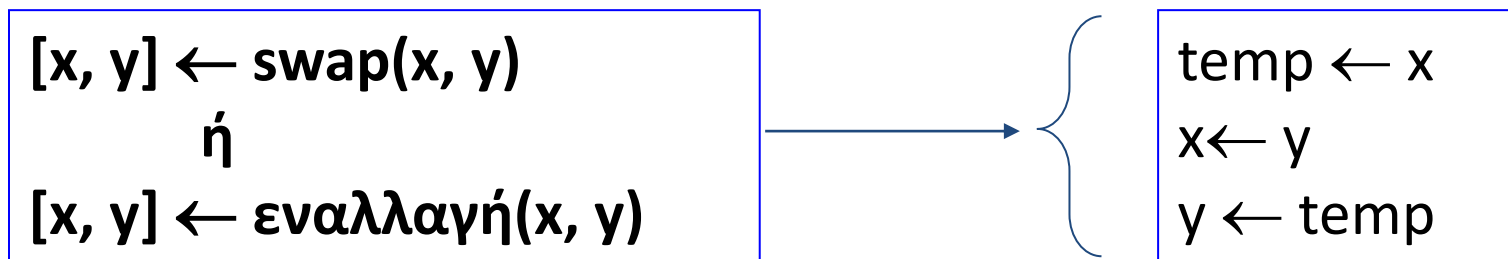


Μερικές από τις μεταβλητές εξόδου και μερικές από τις μεταβλητές εισόδου μπορεί να παραλείπονται

$$\alpha \leftarrow \text{αλγ}(x_1, x_2, \dots, x_n)$$

Ψευδοκώδικας – [5]

Μερικοί χρήσιμοι αλγόριθμοι – συναρτήσεις



$x \leftarrow \text{div}(a, b)$ (άγνωστος προς το παρόν ψευδοκώδικας)

$x \leftarrow \text{mod}(a, b)$ (άγνωστος προς το παρόν ψευδοκώδικας)

Ψευδοκώδικας – [6]

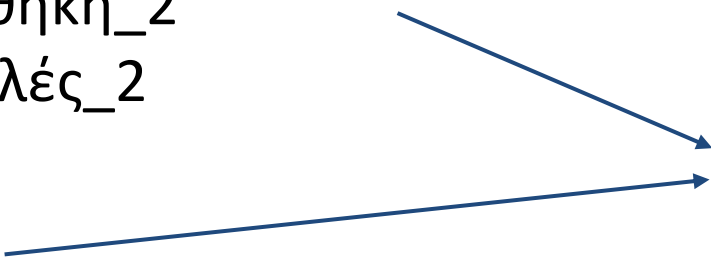
Προσοχή !!!

- ❑ Στις εσοχές: Δηλώνουν πότε τελειώνει μια και πότε αρχίζει άλλη εντολή
- ❑ Στις δεσμευμένες λέξεις : γράφονται **bold**

Η εντολή επιλογής *if...elseif...else...*

```
if συνθήκη_1
    εντολές_1
elseif συνθήκη_2
    εντολές_2
    ...
else
    εντολές_n
```

Προαιρετικά
τμήματα



Ψευδοκώδικας

Παράδειγμα. Γράψτε έναν αλγόριθμο σε ψευδοκώδικα ο οποίος να υπολογίζει την απόλυτη τιμή ενός αριθμού

Ψευδοκώδικας – [7]

Αλγόριθμος: AbsoluteValue1

Είσοδος: x

Έξοδος: a

1	if $x \geq 0$
2	$a \leftarrow x$
3	else
4	$a \leftarrow -x$

Αλγόριθμος: AbsoluteValue2

Είσοδος: x

Έξοδος: a

1	$a \leftarrow x$
2	if $x < 0$
3	$a \leftarrow -x$

Ψευδοκώδικας – [8]

Παράδειγμα. Γνωρίζοντας ότι οι 4 πρώτοι αριθμοί 0, 1, 2, 3, στο δυαδικό σύστημα αρίθμησης είναι 00, 01, 10, 11 να γραφεί αλγόριθμος υπολογισμού αθροίσματος 3 δυφίων (ψηφία του δυαδικού συστήματος αρίθμησης).

Λύση. Έστω x, y, z τα 3 δυφία. Επειδή το άθροισμά τους είναι ένας από τους αριθμούς 00, 01, 10, 11 του δυαδικού συστήματος, θα γράψουμε το αποτέλεσμα στο 2 στοιχείων διάνυσμα s . Δηλ, ο s έχει διάσταση 1×2 .

Κάθε δυφίο παίρνει τιμές 1 ή 0. Άρα **8 περιπτώσεις**

Ψευδοκώδικας – [8]

Παράδειγμα. Γνωρίζοντας ότι οι 4 πρώτοι αριθμοί 0, 1, 2, 3, στο δυαδικό σύστημα αρίθμησης είναι 00, 01, 10, 11 να γραφεί αλγόριθμος υπολογισμού αθροίσματος 3 δυφίων (ψηφία του δυαδικού συστήματος αρίθμησης).

Ψευδοκώδικας – [9]

x	y	z
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Οι 8 περιπτώσεις. Ομαδοποίηση σε 3 περιπτώσεις

Περίπτωση 1

Περίπτωση 2

Περίπτωση 3

Ψευδοκώδικας – [10]

Διάνυσμα 2
στοιχείων

Οι 8 περιπτώσεις ανάγονται
σε 3:

- $x=0$ και $y=0$

- $x=1$ ή $y=1$

- $x=1$ και $y=1$

και σε κάθε μια

εξετάζουμε αν $z=0$ ή $z=1$

Αλγόριθμος : add3bits

Δεδομένα : x, y, z

Αποτελέσματα : s

1	if $x = 0$ and $y = 0$
2	$s \leftarrow [0\ 0]$
3	if $z = 1$, $s(2) = 1$
4	elseif $x = 1$ xor $y = 1$
5	$s \leftarrow [0\ 1]$
6	if $z = 1$, $s \leftarrow [1\ 0]$
7	else
8	$s \leftarrow [1\ 0]$
9	if $z = 1$, $s(2) = 1$

Είσοδος/Εξοδος – [1]

Στοιχειώδης Είσοδος/Εξοδος

`scanf ("%d", &j) ;` -- διαβάζει τον ακέραιο `j`
`scanf ("%f", &a) ;` -- διαβάζει τον πραγματικό `a`
`scanf ("%c", &c) ;` -- διαβάζει τον χαρακτήρα `c`
`scanf ("%d %d %f", &i, &k, &a) ;` -- διαβάζει τους ακέραιους `i` και `k`
και τον πραγματικό `a`, που πληκτρολογήθηκαν με ένα κενό ανάμεσά τους.

`printf ("%d", i) --` εκτυπώνει τον ακέραιο `i`
`printf ("%f", a) --` εκτυπώνει τον πραγματικό `a`
`printf ("%c", c) --` εκτυπώνει τον χαρακτήρα `c`
`printf ("%d %d %f", i, k, a) ;` -- εκτυπώνει τον ακέραιο `i`, τον ακέραιο `k` και τον πραγματικό `a`, αφήνοντας ένα κενό ανάμεσά τους.

Είσοδος/Εξοδος – [3]

Παράδειγμα:

```
int i, k;  
float a;  
char c, name[10];
```

```
scanf("%d", &i)  
scanf("%f", &a)  
scanf("%c", &c)  
scanf("%s", &name)  
scanf("%d%d", &j,&i)
```

```
printf("%d", i)  
printf("%f", a)  
printf("%c", c)  
printf("%s", name)  
printf("%c\n", c)  
printf("\n")
```

Μεταβλητές Εισόδου/Εξόδου

ΟΝΟΜΑ ΑΡΧΕΙΟΥ: 01-add3bits.c

ΜΕΤΑΒΛΗΤΕΣ ΕΙΣΟΔΟΥ:

x – Ακέραιος αριθμός (τύπου int) 0 ή 1.

y – Ακέραιος αριθμός (τύπου int) 0 ή 1.

z – Ακέραιος αριθμός (τύπου int) 0 ή 1.

ΜΕΤΑΒΛΗΤΕΣ ΕΞΟΔΟΥ:

s – Πίνακας δύο θέσεων (τύπου int) στον οποίο αποθηκεύεται το αποτέλεσμα της πρόσθεσης. Το αποτέλεσμα εμφανίζεται στην οθόνη.

Filename: [01-add3bits.c](#)

Πρόβλημα για το σπίτι

Να γραφεί ψευδοκώδικας για τον υπολογισμό των ριζών εξισώσεως 2^{ου} βαθμού με πραγματικούς συντελεστές (θεωρείστε ότι η φανταστική μονάδα είναι αποθηκευμένη στη μεταβλητή j και ότι είναι διαθέσιμος αλγόριθμος υπολογισμού της τετραγωνικής ρίζας).