

Πανεπιστήμιο Αιγαίου

URL: <http://www.aegean.gr>

Ασκήσεις στην Πληροφορική

Μαρίνα Τριπολιτάκη
Πανεπιστήμιο Αιγαίου
Τμήμα Μαθηματικών
832 00 Καρλόβασι
Σάμος



© Copyright Πανεπιστήμιο Αιγαίου, Τμήμα Μαθηματικών
All rights reserved



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

« »

Σελίδα 1 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 2 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Κεφάλαιο 1

Συστήματα αρίθμησης, παραστάσεις, πράξεις

1.1. Συστήματα αρίθμησης

Άσκηση 1.1.1 Μετατρέψτε τους ακόλουθους αριθμούς:

1. Τον $(1010.011)_2$ στο δεκαδικό
2. Τον $(630.4)_8$ στο δεκαδικό
3. Τους $(11101)_2$, $(100110101)_2$ στο οκταδικό.
4. Τον $(1011111000)_2$ στο δεκαεξαδικό.
5. Τον $(1C5A)_{16}$ στο δεκαδικό.

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 3 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Άσκηση 1.1.2 Μετατρέψτε τους κλασματικούς αριθμούς στα συστήματα που σας ζητούνται:

1. Ο $(0.625)_{10}$ στο δυαδικό.
2. Ο $(0.101011)_2$ στο δεκαδικό

Υπόδειξη-Λύση

1.2. Αναπαραστάσεις αριθμών

Άσκηση 1.2.1 Βρείτε τα συμπληρώματα του 1001 ως προς 1 και 2, χρησιμοποιώντας

1. 4 bit πληροφορίας
2. 5 bit πληροφορίας

Υπόδειξη-Λύση

Άσκηση 1.2.2 Βρείτε το συμπλήρωμα ως προς 1 και 2 του προσημασμένου 1001010.

Υπόδειξη-Λύση

Άσκηση 1.2.3 Βρείτε τα συμπληρώματα ως προς 1 και 2 των αριθμών $+(1011)_2$ και $-(1011)_2$ χρησιμοποιώντας 5 και 7 bit πληροφορίας. Προσοχή! Εδώ οι αριθμοί είναι προσημασμένοι.

Υπόδειξη-Λύση

1.3. Πράξεις

Άσκηση 1.3.1 Κάντε τις πράξεις (στο δυαδικό), καθορίζοντας κάθε φορά το κρατούμενο C:

1. $0 + 0$

2. $1 + 0$

3. $0 + 1$

4. $1 + 1$

Άσκηση 1.3.2 Προσθέστε τους δυαδικούς 101011 και 1111.

Άσκηση 1.3.3 Αφαιρέστε τους δυαδικούς

1. $10111 - 101$

2. $101011 - 1111$.

Άσκηση 1.3.4 Κάντε την αφαίρεση $23_{10} - 4_{10}$, με χρήση του συμπληρώματος ως προς 1.

Υπόδειξη-Λύση

Άσκηση 1.3.5 Κάντε την αφαίρεση $68 - 54$ με χρήση συμπληρώματος ως προς 1. **Υπόδειξη-Λύση**

Άσκηση 1.3.6 Κάντε την αφαίρεση $68 - 54$ με χρήση συμπληρώματος ως προς 2. **Υπόδειξη-Λύση**

Άσκηση 1.3.7 Κάντε την αφαίρεση $18 - 115$ σε παράσταση συμπληρώματος ως προς 1. **Υπόδειξη-Λύση**

Άσκηση 1.3.8 Κάντε την αφαίρεση $18 - 115$ σε παράσταση συμπληρώματος ως προς 2. **Υπόδειξη-Λύση**

Υπόδειξη-Λύση

Υπόδειξη-Λύση

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 4 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 5 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Κεφάλαιο 2

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

2.1. Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Άσκηση 2.1.1 Δείξτε ότι οι λογικές συναρτήσεις $f = \bar{x}_1\bar{x}_2 + \bar{x}_1x_2 + x_1x_2$ και $g = \bar{x}_1 + x_2$ είναι ίσες, με τη βοήθεια των αξιωμάτων της άλγεβρας Boole.

Υπόδειξη-Λύση

Άσκηση 2.1.2 Βρείτε τον πίνακα αληθείας της συνάρτησης f της άσκησης 2.1.1

Υπόδειξη-Λύση

Άσκηση 2.1.3 Υπολογίστε τον πίνακα αληθείας της συνάρτησης

$$f = \bar{x}_1\bar{x}_2x_3 + x_1\bar{x}_2\bar{x}_3 + x_1\bar{x}_2x_3 + x_1x_2\bar{x}_3.$$

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 6 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 2.1.4 Βρείτε τη συνάρτηση που αντιστοιχεί στον παρακάτω πίνακα αληθείας.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Υπόδειξη-Λύση

Άσκηση 2.1.5 Υπολογίστε τη συνάρτηση f :



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



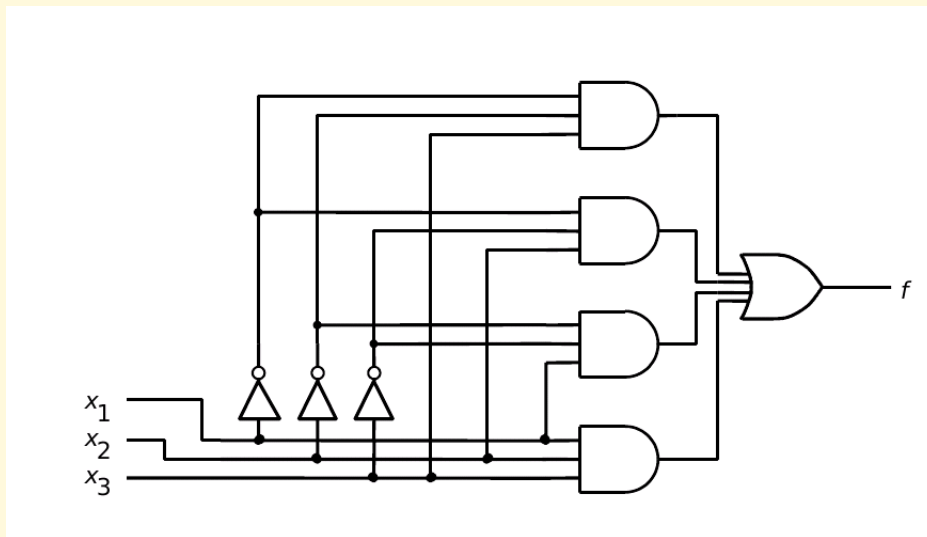
Σελίδα 7 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Άσκηση 2.1.6 Ελαχιστοποιήστε τις συναρτήσεις

$$f = x\bar{y} + xy + \bar{x}y, g = \bar{x}\bar{y} + x\bar{y} + \bar{x}\bar{y}.$$

Υπόδειξη-Λύση

Άσκηση 2.1.7 Ελαχιστοποιήστε τη συνάρτηση

$$f = x\bar{y}z + x\bar{y}\bar{z} + xy\bar{z}.$$

Υπόδειξη-Λύση

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 8 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 2.1.8 Ελαχιστοποιήστε τη συνάρτηση

$$f(x, y, z, w) = \sum (0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14)$$

Υπόδειξη-Λύση

Άσκηση 2.1.9 (Ημαδριστής) Προσθέτοντας δύο μονοψήφιους δυαδικούς αριθμούς, το αποτέλεσμα της πράξης μπορεί να είναι μονοψήφιο ή διψήφιο. Μπορεί να έχει κρατούμενο 0 ή ένα. Συμβολίζουμε με C το κρατούμενο και με S «άδρυσμά» τους, κρατώντας μόνο το ψηφίο των μονάδων. Πχ το S στην περίπτωση του αδροίσματος των 1, 1 είναι 0 και το $C = 1$. Στην περίπτωση των 1, 0 είναι $S = 1$ και $C = 0$.

Υλοποιήστε την παραπάνω διαδικασία με λογικό κύκλωμα δύο μεταβλητών x, y με δύο εξόδους C, S .

Υπόδειξη-Λύση

Άσκηση 2.1.10 (Πλήρης αδροιστής) Αναπαραστήστε με λογικό κύκλωμα τριών εισόδων και δύο εξόδων την πρόσθεση τριών δυαδικών αριθμών. Οι έξοδοι S, C είναι αντίστοιχα το «άδρυσμα» (εδώ το πλήθος των μονάδων) και το κρατούμενο.

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 9 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Κεφάλαιο 3

Αλγόριθμοι

3.1. Αλγόριθμοι

Άσκηση 3.1.1 Γράψτε ψευδοκώδικα που

1. διαβάξει άγνωστο πλήθος αριθμών και
2. υπολογίζει και τυπώνει τον μέσο όρο των αριθμών που διάβασε.

Υπόδειξη-Λύση

Άσκηση 3.1.2 Να γράψετε αλγόριθμο που να διαβάξει θετικό διψήφιο ακέραιο, να αντι-μεταδέτει τα ψηφία του και να τυπώνει τον αριθμό που προκύπτει.

Υπόδειξη-Λύση

Άσκηση 3.1.3 Να ταξινομηθούν 10 ακέραιοι με φθίνουσα σειρά με τη μέθοδο της φυσαλίδας.

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 10 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Κεφάλαιο 4

Fortran

4.1. Εισαγωγικά

Άσκηση 4.1.1 Υπολογίστε τις τιμές των παραστάσεων στη Fortran.

1. $7 + 5 * 4 + 8$

2. $7 + 5/4 + 8$

3. $7 + 5/4.0 + 8$

4. $1/2$

5. $1.0/2.0$

Υπόδειξη-Λύση

Άσκηση 4.1.2 Σωστό ή λάθος;

1. Xyz
2. $_abc$
3. 9_count
4. $A_$
5. $m.t$
6. $m-t$

Υπόδειξη-Λύση

Άσκηση 4.1.3 Γράψτε πρόγραμμα που να εκτυπώνει στην οθόνη το μήνυμα Hello word!
Υπόδειξη-Λύση

Άσκηση 4.1.4 Γράψτε πρόγραμμα στη Fortran στο οποίο

1. δηλώνεται ακέραια μεταβλητή με όνομα x
2. δηλώνεται πραγματική μεταβλητή με όνομα y
3. δηλώνεται πραγματική μεταβλητή διπλής ακρίβειας με όνομα z
4. ανατίθεται στην x η τιμή 2
5. ανατίθεται στην y η τιμή 5.1
6. ανατίθεται στην z η τιμή 4.52
7. τυπώνεται στην οθόνη η τιμή της $x + y/2 * \sin(z)$



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 11 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Υπόδειξη-Λύση

Άσκηση 4.1.5 Όταν οι τελεστές +, -, */ εμφανίζονται μεταξύ αριθμών διαφορετικού τύπου μπορεί να έχουμε απώλεια ακρίβειας. Για να το διαπιστώσετε, μελετήστε και τρέξτε το παρακάτω πρόγραμμα:

Υπόδειξη-Λύση

Άσκηση 4.1.6 Ο αλγόριθμος του Gauss για τον υπολογισμό της ημερομηνίας του Πάσχα των ορθοδόξων χριστιανών σε συγκεκριμένο έτος (μέχρι το 2099) είναι ο εξής:

1. Θεωρούμε ως δεδομένο εισόδου το έτος που μας ενδιαφέρει.
2. Ορίζουμε κάποιες ακέραιες ποσότητες σύμφωνα με τους ακόλουθους τύπους:
3. $r1 = \text{υπόλοιπο διαίρεσης του έτους με το } 19.$
4. $r2 = \text{υπόλοιπο διαίρεσης του έτους με το } 4.$
5. $r3 = \text{υπόλοιπο διαίρεσης του έτους με το } 7.$
6. $ra = 19r1 + 16.$
7. $r4 = \text{υπόλοιπο διαίρεσης του } ra \text{ με το } 30.$
8. $rb = 2r2 + 4r3 + 6r4.$
9. $r5 = \text{υπόλοιπο διαίρεσης του } rb \text{ με το } 7.$
10. $rc = r4 + r5.$

Το rc είναι το πλήθος των ημερών μετά την 3η Απριλίου του συγκεκριμένου έτους, της ημερομηνίας του Πάσχα. Γράψτε σε Fortran τον παραπάνω αλγόριθμο. Φροντίστε να τυπώνει κατάλληλο μήνυμα και τον αριθμό rc αφού τον υπολογίσει.

Υπόδειξη-Λύση

Άσκηση 4.1.7 Γράψτε πρόγραμμα που να ανταλλάσσει τις τιμές των μεταβλητών x, y .

Υπόδειξη-Λύση

Συστήματα αρίθμησης

Κυκλώματα, λογικές συν-
ναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 12 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 13 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

4.2. Εντολές ελέγχου

4.2.1. if-then-else

Η εντολή IF είναι μία από τις βασικές δομές διακλάδωσης της γλώσσας. Ελέγχει τη ροή του κώδικα με τη λογική τιμή condition. Αν η condition είναι αληθής εκτελείται το blockA διαφορετικά εκτελείται το blockB. Η σύνταξη της εντολής είναι η εξής:

```
IF (condition) THEN
    ...
    // block A
ELSE
    ...
    // block B
END IF
```

Άσκηση 4.2.1 Γράψτε ένα πρόγραμμα που με είσοδο από τον χρήστη να ένα θετικό πραγματικό r να υπολογίζει και να εκτυπώνει τον όγκο V μιας σφαίρας, βάση του τύπου $V = \frac{4}{3}\pi r^3$.
Υπόδειξη-Λύση

Άσκηση 4.2.2 Γράψτε πρόγραμμα που να υπολογίζει το μεγαλύτερο αριθμό μεταξύ των x, y .
Υπόδειξη-Λύση

Άσκηση 4.2.3 Γράψτε πρόγραμμα που να υπολογίζει τον μικρότερο αριθμό μεταξύ των a, b .
Υπόδειξη-Λύση

Άσκηση 4.2.4 Γράψτε πρόγραμμα που να υπολογίζει τον μικρότερο αριθμό μεταξύ των a, b, c .
Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 14 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 4.2.5 Γράψτε ένα πρόγραμμα που να υπολογίζει την απόλυτη τιμή πραγματικού αριθμού.
Υπόδειξη-Λύση

Άσκηση 4.2.6 Σε κάθε αγοραπωλησία μετοχών, η χρηματιστηριακή εταιρία εισπράττει προμήθια σύμφωνα με τον ακόλουθο πίνακα:

Αξία μετοχών σε ευρώ	Προμήθεια σε ευρώ
Κάτω από 2500	$30 + 1.7\%$
2500-6000	$40 + 0.6\%$
6000-20000	$60 + 0.2\%$
20000-100000	$100 + 0.11\%$
Πάνω από 100000	$150 + 0.01\%$

Γράψτε πρόγραμμα που να υπολογίζει την προμήθεια.

Υπόδειξη-Λύση

Άσκηση 4.2.7 Γράψτε πρόγραμμα που να υπολογίζει τις ρίζες δευτεροβάθμιας εξίσωσης.
Υπόδειξη-Λύση

4.2.2. select case

Η σύνταξη της select case είναι:

```
SELECT CASE (i)
  CASE (value1)
    ...
  CASE (value2)
    ...
  ...
  CASE (valueN)
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 15 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

```
...
CASE default
...
END SELECT
```

Η τιμή i είναι ακέραιου τύπου ή λογική ή χαρακτήρας. Οι τιμές $value_1, value_2, \dots, value_N$

συγκρίνονται με την τιμή ελέγχου i , και εκτελείται το αντίστοιχο μπλοκ εντολών. Αν δεν επαληθεύεται καμία από αυτές τότε εκτελούνται οι εντολές που αντιστοιχούν στο μπλοκ εντολών της default.

Άσκηση 4.2.8 Γράψτε πρόγραμμα που να εκτυπώνει το πρόσημο του ακεραίου x . (Με χρήση του select case.) **Υπόδειξη-Λύση**

Άσκηση 4.2.9 Γράψτε πρόγραμμα που να μετατρέπει τους βαθμούς Κελσίου C σε Φαρενάιτ F και αντίστροφα. Η μετατροπή θα πρέπει να γίνεται με τη βοήθεια συναρτήσεων. Ο χρήστης θα επιλέγει αρχικά το είδος της μετατροπής (τι θα μετατρέψει σε τι). **Υπόδειξη-Λύση**

Άσκηση 4.2.10 Γράψτε πρόγραμμα που να υπολογίζει το $n!$ **Υπόδειξη-Λύση**

4.2.3. Γενικές ασκήσεις

Άσκηση 4.2.11 Γράψτε τους αριθμούς a, b, c σε αύξουσα σειρά. **Υπόδειξη-Λύση**

Άσκηση 4.2.12 Έστω x ο βαθμός στο μάθημα της φυσικής. Αν $x < 5$ τότε το γραμματικό του ισοδύναμο grade είναι f , διαφορετικά αν $5 \leq x < 6$ τότε $grade=d$, διαφορετικά αν $6 \leq x < 7$ τότε $grade=c$, διαφορετικά αν $7 \leq x < 8$ τότε $grade=b$, διαφορετικά αν $8 \leq x$ τότε $grade=a$. **Υπόδειξη-Λύση**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 16 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 4.2.13 Που βρίσκεται το θάθος στο παρακάτω πρόγραμμα ;;

```
program minimum
  real :: a, b, minimum
  read*, a, b
  if (a <= b) then
    minimum = a
  else
    minimum = b
  end if
  print*, 'To minimum tw'n ', a, ' kai ', &
    b, ' e'ina'i ', minimum
end program minimum
```

Υπόδειξη-Λύση

Άσκηση 4.2.14 Υπολογίστε το άθροισμα $\sum_{i=0}^{10} 3^{i+1}$

Υπόδειξη-Λύση

Άσκηση 4.2.15 Υπολογίστε το άθροισμα $\sum_{i=0}^{100} \sum_{j=0}^{200} \frac{1}{i+j+1}$.

Υπόδειξη-Λύση

Άσκηση 4.2.16 Γράψτε πρόγραμμα που να υπολογίζει το άθροισμα και το πλήθος των όρων της σειράς $1 + 1/2^2 + 1/3^2 + \dots + 1/n^2 + \dots$ μέχρι του όρου που είναι μικρότερος από $\epsilon = 0.0005$

Υπόδειξη-Λύση

Άσκηση 4.2.17 Να υπολογιστεί το άθροισμα

$$1 - \frac{1}{2!} + \frac{1}{3!} - \frac{1}{4!} + \dots + (-1)^{n+1} \frac{1}{n!} + \dots$$

μέχρι ο $\frac{1}{n!}$ να γίνει μικρότερος του ϵ , για δοσμένο ϵ . Πόσοι όροι απαιτούνται;

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 17 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 4.2.18 Να γραφεί πρόγραμμα που να διαβάζει η ακέραιους και να υπολογίζει

1. Το άθροισμα των τετραγώνων των αρνητικών
2. Το άθροισμα των περιπών που είναι πολλαπλάσιοι του 3
3. Το άθροισμα των περιπών που είναι πολλαπλάσιοι του 3 και πολλαπλάσιοι του 5.

Υπόδειξη-Λύση

Άσκηση 4.2.19 Να γραφεί πρόγραμμα που να μετατρέπει ακέραιο σε δυαδικό. **Υπόδειξη-Λύση**

Άσκηση 4.2.20 Γράψτε πρόγραμμα που να υπολογίζει το πλήθος των δεκαδικών ψηφίων ενός ακεραίου. **Υπόδειξη-Λύση**

Άσκηση 4.2.21 Είναι γνωστό ότι η ακολουθία a_n με

$$a_0 = 1, a_{n+1} = \frac{1}{2} \left(a_n + \frac{a}{a_n} \right), n = 0, 1, 2, \dots$$

συγκλίνει στο \sqrt{a} , για $a > 0$. Γράψτε ένα πρόγραμμα που υπολογίζει προσεγγίσεις του \sqrt{a} , $a > 0$. **Υπόδειξη-Λύση**

Άσκηση 4.2.22 Γράψτε πρόγραμμα που να δημιουργεί το ιστόγραμμα δοσμένων αριθμών x_1, \dots, x_n για τα διαστήματα $(-\infty, 0]$, $(0, 10]$, $(10, 20]$ και $(20, \infty)$.

Παράδειγμα, αν δώσουμε τους αριθμούς $-20, 40, 3.4, 10.23, 3, 7, 12, 55, 122, 74$ θα πρέπει να εκτυπωθεί στην οθόνη το ιστόγραμμα

$(-\infty, 0]$: *
 $(0, 10]$: ***
 $(10, 20]$: **
 $(20, \infty)$: ****



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 18 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη-Λύση

Άσκηση 4.2.23 Γράψτε πρόγραμμα που να εκτυπώνει το ακόλουθο. Το πλήθος των γραμμών n δίνεται από τον χρήστη κατά την εκτέλεση του προγράμματος (στο σχήμα $n = 10$).

```
*
***
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

Υπόδειξη-Λύση

Άσκηση 4.2.24 Γράψτε ένα πρόγραμμα που με είσοδο από τον χρήστη ένα θετικό ακέραιο $n \geq 3$ να εκτυπώνει ένα τετράγωνο πλαίσιο με αστερίσκους, όπως φαίνεται στο παράδειγμα για $n = 4$:

```
****
* *
* *
****
```

Υπόδειξη-Λύση

4.3. Εντολές επανάληψης (do)

Η βασική σύνταξη της Do είναι η ακόλουθη



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 19 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
DO i=a, b, n
```

```
.....
```

```
END DO
```

όπου i είναι ακέραια μεταβλητή που παίρνει τις τιμές $a, a + n, a + 2n, \dots, b$. Το n μπορεί να παραλείπεται. Αν παραλείπεται τότε θεωρείται ότι είναι το 1.

Παρακάτω θα δείτε διάφορες μορφές που παίρνει η DO.

Άσκηση 4.3.1 Γράψτε πρόγραμμα που να υπολογίζει τους ακέραιους από το 1 ως το 20.
Υπόδειξη-Λύση

Άσκηση 4.3.2 Γράψτε πρόγραμμα που να ζητά από τον χρήστη την εισαγωγή χαρακτήρα. Το πρόγραμμα θα πρέπει να τερματίζει μόνο στην περίπτωση που ο εισαγόμενος χαρακτήρας είναι ο '0'.
Υπόδειξη-Λύση

Άσκηση 4.3.3 Γράψτε πρόγραμμα που να εκτυπώνει το ακόλουθο. Το πλήθος των γραμμών n δίνεται από τον χρήστη κατά την εκτέλεση του προγράμματος (στο σχήμα $n = 10$).

```
*
**
***
****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

«

»

«

»

Σελίδα 20 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Άσκηση 4.3.4 Γράψτε πρόγραμμα που να ζητά από τον χρήστη μία σειρά από πραγματικούς θετικούς. Το πλήθος τους δεν είναι γνωστό εκ των προτέρων. Το διάβασμα των τιμών σταματά όταν ο χρήστης εισάγει αρνητικό αριθμό. Το πρόγραμμα θα υπολογίζει τον μέσο όρο των αριθμών αυτών. **Υπόδειξη-Λύση**

4.4. Πίνακες

Οι παρακάτω δηλώσεις

```
real,      dimension(10)      :: a
integer,   dimension(2:3)     :: b
character, dimension(4)       :: c
```

αναφέρονται αντιστοίχως, σε ένα πίνακα $a \in \mathbb{R}^{10}$ (δηλ. σε διάνυσμα μήκους 10), σε ένα πίνακα $b \in \mathbb{R}^{n \times m}$ (με n γραμμές και m στήλες) και σε ένα διάνυσμα μήκους 4 όπου κάθε στοιχείο του είναι χαρακτήρας (σε μία «λέξη» δηλαδή).

Ένας πίνακας δηλαδή είναι μία συλλογή στοιχείων ίδιου τύπου.

Άσκηση 4.4.1 Έστω a, b πραγματικά διανύσματα (array) μήκους n . Τι σημαίνουν οι παρακάτω δηλώσεις;

1. $b = a + 1$
2. $b = a * 2$
3. $b = a/c$
4. $b = 1.0$
5. $b == a$

6. $b = a$

7. $b = \cos(a)$

Υπόδειξη-Λύση

Άσκηση 4.4.2 Γράψτε πρόγραμμα που να διαβάζει δύο διανύσματα του \mathbb{R}^3 και να υπολογίζει το εσωτερικό τους γινόμενο.

Υπόδειξη-Λύση

Άσκηση 4.4.3 Γράψτε πρόγραμμα που να διαβάζει διάνυσμα μήκους 10 και να υπολογίζει το άθροισμα των στοιχείων του.

Υπόδειξη-Λύση

Άσκηση 4.4.4 Γράψτε πρόγραμμα που να ορίζει πίνακα A 4×4 ως εξής: το στοιχείο $A_{i,j}$ που βρίσκεται στην i γραμμή και στη j στήλη να είναι το $i + j$. Βρείτε το ίχνος αυτού του πίνακα.

Υπόδειξη-Λύση

Άσκηση 4.4.5 Γράψτε πρόγραμμα που να ζητάει από τον χρήστη την εισαγωγή δύο πινάκων 2×2 και να υπολογίζει το γινόμενό τους.

Υπόδειξη-Λύση

Άσκηση 4.4.6 Να βρεθεί το μέγιστο και το ελάχιστο στοιχείο ενός πίνακα με n στοιχεία.
Υπόδειξη-Λύση

4.4.1. Arrays allocation

Η δήλωση

```
real, dimension(:, :), allocatable :: myarray
```

χρησιμοποιείται όταν θέλουμε να ορίσουμε μέσα στο σώμα του προγράμματος τη διάσταση του $n \times m$ πίνακα `myarray`. Η διάστασή του $n \times m$ δεν καθορίζεται κατά τη δήλωσή του. Αφήνεται να καθοριστεί στη συνέχεια, ανάλογα με τις ανάγκες του προγράμματος.

Άσκηση 4.4.7 Γράψτε πρόγραμμα που να μετατρέπει ακεραίους σε δυαδικούς.
Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 21 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 22 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

4.5. Συναρτήσεις, υπορουτίνες

Οι συναρτήσεις στα μαθηματικά έχουν ένα πεδίο ορισμού και ένα σύνολο τιμών: $f : A \rightarrow B, x \mapsto f(x)$. Το x ονομάζεται όρισμα της f .

Στη Fortran κάθε συνάρτηση έχει τη μορφή

```
type function function_name(x1, x2, ...)
type :: x1
type :: x2
.....
return
end function function_name
```

Πχ. η μαθηματική συνάρτηση $f : \mathbb{Z} \rightarrow \mathbb{R}, n \mapsto \sin(n)$ γράφεται ως εξής

```
real function mysin(n)
integer :: n

mysin=sin(n)

return

end function mysin
```

Για την υπορουτίνα που δεν επιστρεφει τιμή (δεν έχει πεδίο τιμών δηλαδή) η σύνταξη είναι

```
subroutine name(x1, x2, ...)
type :: x1
type :: x2
```

.....

end subroutine name

Δείτε τα επόμενα παραδείγματα.

- Άσκηση 4.5.1** 1. Μετατρέψτε σε συνάρτηση της Fortran την μαθηματική συνάρτηση $f(x) = \sin(x^2)$, $x \in \mathbb{Z}$.
2. Γράψτε πρόγραμμα που να κάνει χρήση της παραπάνω άσκησης.

Υπόδειξη-Λύση

- Άσκηση 4.5.2** Μετατρέψτε τη μαθηματική συνάρτηση

$$f(x, y) = \begin{cases} x - y & \text{αν } x > y \\ x^2 & \text{αλλιώς} \end{cases}$$

σε συνάρτηση της Fortran.

Υπόδειξη-Λύση

- Άσκηση 4.5.3** Γράψτε συνάρτηση που να υπολογίζει το $n!$. Να την καλέσετε για να υπολογίσετε τα $n!$, για $n = 1, \dots, 10$.

Υπόδειξη-Λύση

- Άσκηση 4.5.4** Γράψτε συνάρτηση logical function `iseven(i)` που να επιστρέφει την τιμή `.TRUE.` αν ο i είναι άρτιος και `.FALSE.` διαφορετικά. Γράψτε πρόγραμμα που να κάνει χρήση της παραπάνω συνάρτησης και να εκτυπώνει όλους τους άρτιους από το 1 στο 10.

Υπόδειξη-Λύση

- Άσκηση 4.5.5** Τροποποιήστε την άσκηση 4.1.7 ώστε την αλλαγή των τιμών των μεταβλητών να την πραγματοποιεί υπορουτίνα.

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 23 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

⏪ ⏩

◀ ▶

Σελίδα 24 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Άσκηση 4.5.6 Δημιουργήστε μία συνάρτηση που να δέχεται ως όρισμα ακέραιους και να επιστρέφει 1 αν ο ακέραιος είναι πρώτος και 0 αλλιώς. Ένας ακέραιος n ονομάζεται πρώτος, αν δεν είναι ± 1 και οι μοναδικοί διαιρέτες του είναι οι $\pm 1, \pm n$. **Υπόδειξη-Λύση**

Άσκηση 4.5.7 Η σειρά

$$\sin(x) = x - x^3/3! + x^5/5! - \dots$$

χρησιμοποιείται για τον υπολογισμό του $\sin x$ (το x εκφρασμένο σε rad). Χρησιμοποιείστε την παραπάνω σειρά για να υπολογίσετε το $\sin 35^\circ$. Συγκρίνετε το αποτέλεσμα με το αποτέλεσμα που δίνει η κλήση της συνάρτησης \sin του συστήματος. **Υπόδειξη-Λύση**

Άσκηση 4.5.8 Ποιά είναι το λάθος στο παρακάτω πρόγραμμα:

```
program wrong
  implicit none
  .....
contains
  integer function internalfunction(.....)
    implicit none
    .....
contains
  real function funct(.....)
    implicit none
    .....
  end function funct
end function internalfunction
end program wrong
```

Υπόδειξη-Λύση

Άσκηση 4.5.9 Γράψτε μία ρουτίνα η οποία να επιστρέφει τη σειρά των στοιχείων ενός πίνακα ακεραίων. **Υπόδειξη-Λύση**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 25 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

4.5.1. Αναδρομικές συναρτήσεις

Άσκηση 4.5.10 Υπολογίστε το $n!$ με συνάρτηση αναδρομής.

Υπόδειξη-Λύση

4.6. Συναρτήσεις (2)

Άσκηση 4.6.1 Γράψτε συνάρτηση που να παίρνει ως ορίσματα δύο arrays και να επιστρέφει ένα array όπου το i -στοιχείο του είναι το maximum των αντίστοιχων στοιχείων των ορισμάτων.

Υπόδειξη-Λύση

Άσκηση 4.6.2 Γράψτε υπορουτίνα που να παραγωγίζει πολυώνυμο (οποιοδήποτε βαθμού) και υπορουτίνα που να υπολογίζει την τιμή πολυωνύμου στο x_0 . Δημιουργήστε ένα πρόγραμμα που να επιδεικνύει τα παραπάνω.

Υπόδειξη-Λύση

Άσκηση 4.6.3 Γράψτε συνάρτηση που να δέχεται ως είσοδο τρεις πραγματικούς αριθμούς και να επιστρέφει 1 αν οι αριθμοί αυτοί είναι μήκη πλευρών τριγώνου και 0 αλλιώς. Στην περίπτωση που οι αριθμοί σχηματίζουν τρίγωνο, να απαντάται αν το τρίγωνο είναι ισόπλευρο, ισοσκελές ή σκαληνό.

Υπόδειξη-Λύση

Άσκηση 4.6.4 Γράψτε υπορουτίνα $\text{inverse}(a, b, \text{det})$ (και πρόγραμμα για επίδειξη) που να δέχεται ως όρισμα τον 2×2 πίνακα a , να επιστρέφει την ορίζουσα του det και τον αντίστροφό του -αν υπάρχει, στον b .

Υπόδειξη-Λύση

Άσκηση 4.6.5 Για την δήλωση μίας χρονολογίας συχνά χρησιμοποιείται η μορφή: $yyyy\text{mm}dd$, όπου τα τέσσερα πρώτα ψηφία $yyyy$ δηλώνουν το έτος, τα επόμενα δύο mm δηλώνουν τον μήνα, και τα μεθεπόμενα dd δηλώνουν την ημέρα. Πχ η χρονολογία 12730206 δηλώνει αναφέρεται στις 6/2/1973. Γράψτε πρόγραμμα που να διαβάξει ακεραίους της μορφής $yyyy\text{mm}dd$ και να εκτυπώνει τη χρονολογία στη μορφή



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

« « » »

◀ ▶

Σελίδα 26 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Year = yyyy

Month = mm

Day = dd

δηλαδή, στο παράδειγμά μας θα πρέπει να εκτυπώνει:

Year = 1973

Month = 2

Day = 6

Υπόδειξη-Λύση

Το πρόγραμμα θα ζητάει από τον χρήστη την εισαγωγή χρονολογίας (στη μορφή yyyymmdd), και θα τερματίζει μόνο αν ο χρήστης δώσει τον αριθμό 0.

Το έτος 1973 είναι το (ακέραιο) πηλίκο του 12730206 με το 10000 (δηλ. το $12730206/10000$).

Για να υπολογίσουμε τον μήνα, αρχικά βρίσκουμε το υπόλοιπο της διαίρεσης του 12730206 με το 100 ($\text{mod}(\text{number}, 10000)$) και στη συνέχεια υπολογίζουμε το πηλίκο της διαίρεσης του υπολοίπου αυτού με το 100. Για να υπολογίσουμε την ημέρα, αρκεί να υπολογίσουμε το πηλίκο του 12730206 με το 100.

Άσκηση 4.6.6 Γράψτε πρόγραμμα που να υπολογίζει τον μέγιστο κοινό διαιρέτη δύο ακέραιων.

Υπόδειξη-Λύση

Άσκηση 4.6.7 Γράψτε πρόγραμμα που να υπολογίζει με τη μέθοδο του Bolzano ρίζα της εξίσωσης $f(x) = 0$: Αν η συνάρτηση $f(x)$ είναι συνεχής στο διάστημα $[a, b]$ και $f(a)f(b) < 0$ τότε υπάρχει $x_0 \in [a, b]$ με $f(x_0) = 0$.

Για παράδειγμα, μπορείτε να χρησιμοποιήσετε τη συνάρτηση $f(x) = x^3$.

Υπόδειξη-Λύση

Άσκηση 4.6.8 Χρησιμοποιήστε τη μέθοδο του τραπεζίου για να ολοκληρώσετε συναρτήσεις ορισμένες σε κλειστά διαστήματα.

Υπόδειξη-Λύση



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 27 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Άσκηση 4.6.9 (Bubble sorting) Να ταξινομηθούν με τη μέθοδο της φουσαλίδας κατά φθίνουσα σειρά 10 ακέραιοι οι οποίοι εισάγονται από τον χρήστη κατά την εκτέλεση του προγράμματος.

Υπόδειξη-Λύση

4.7. Αρχεία

Άσκηση 4.7.1 Γράψτε πρόγραμμα που να διαβάζει από ένα αρχείο με όνομα student.records τα ονόματα και τους βαθμούς των φοιτητών σε 3 μαθήματα και να αποθηκεύει σε αρχείο με όνομα student.scores τον σταθμισμένο μέσο όρο των βαθμών αυτών. Τα βάρη ας είναι αντιστοίχως στα μαθήματα αυτά ίσα προς .3, .3 και .4. Πχ, αν ο x φοιτητής πέτυχε βαθμολογία 10, 5 και 6 στο πρώτο, δεύτερο και τρίτο μάθημα αντιστοίχως τότε, ο σταθμισμένος μέσος όρος είναι $10 \times .3 + 5 \times .3 + 6 \times .4 == 6.9$. Το student.records ας έχει τη μορφή

```
Tripolitaki, Marina    5 7 3
```

Υπ'οδείξη-Λύση

Άσκηση 4.7.2 Γράψτε συνάρτηση που να μετρά το πλήθος των εμφανίσεων ενός ακέραιου αριθμού, που θα δίνεται ως όρισμα, στο αρχείο numbers, η πρώτη γραμμή του οποίου θα περιέχει το πλήθος των αριθμών που ακολουθούν. Καλέστε τη συνάρτηση για να υπολογίσετε το πλήθος των εμφανίσεων των αριθμών 1234 και 234. Το αρχείο numbers ας είναι το



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 28 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη-Λύση

12
1234
23981
123823123
2312
1234
234
0
456
4324
1234
1234
0



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

⏪ ⏩

◀ ▶

Σελίδα 29 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Βιβλιογραφία

- [1] ITS. An introduction to programming in fortran 90, Durham University, 2007.
- [2] W. Press et all. Numerical recipes in fortran 77, Cambridge University Press, 1992.
- [3] Morris Mano. Ψηφιακή Σχεδίαση, Παπασωτηρίου, 1992.
- [4] Michael Metcalf. Fortran 90/95 explained, Oxford university press, 1999.
- [5] Σ. Σταματιάδης. Υλεκτρονικοί Υπολογιστές I: Μία εισαγωγή στη γλώσσα προγραμματισμού Fortran 95, <http://www.csd.uoc.gr>, 2008.



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 30 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Αποδείξεις



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 31 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Έστω x κλασματικός δεκαδικός και y το δυαδικό αντίστοιχο του. Για να τον μετατρέψουμε σε δυαδικό, εργαζόμαστε ως εξής:

1. Πολλαπλασιάζουμε τον x με 2. Έστω A το ακέραιο του μέρους και K το κλασματικό του $2x$. Το A είναι το αριστερότερο κλασματικό ψηφίο του y (το πρώτο ψηφίο μετά την υποδιαστολή).
2. Πολλαπλασιάζουμε τον κλασματικό του παραπάνω βήματος με 2 και παίρνουμε ένα νέο ακέραιο μέρος A και ένα νέο κλασματικό K . Γράφουμε το A στα δεξιά του δυαδικού κλασματικού που βρήκαμε στο προηγούμενο βήμα (το δεύτερο ψηφίο μετά την υποδιαστολή είναι το A).
3. Επαναλαμβάνουμε το παραπάνω βήμα, μέχρις ότου το K γίνει μηδέν.

□

Λύση

Πίσω στην Άσκηση 1.1.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 32 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Αρχικά μετασχηματίζουμε τον 1001 σε ισοδύναμο χρησιμοποιώντας 4 και 5 bit: 1001 και 01001. Στο συμπλήρωμα ως προς 1 απλά μετατρέπουμε τα 1 σε 0 και αντίστροφα. Στο συμπλήρωμα ως προς 2 προσθέτουμε στο παραπάνω (συμπλήρωμα ως προς 1) το 1.

□

Λύση

Πίσω στην Άσκηση 1.2.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 33 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Το πιο αριστερό bit είναι το 1, το οποίο αντιστοιχεί στο πρόσημο «-». Το μέτρο του 1001010 είναι το 001010. Βρίσκουμε το συμπλήρωμα του μέτρου ως προς 1 και τοποθετούμε σε αυτό ένα 1 στα αριστερά του, ώστε να τον μετατρέψουμε σε αρνητικό. Αυτό που θα πάρουμε είναι το συμπλήρωμα του 1001010 ως προς 1. Ομοίως εργαζόμαστε και για το συμπλήρωμα του 1001010 ως προς 2. Δες επίσης και την υπόδειξη της άσκησης **1.2.3**

□

Λύση

Πίσω στην Άσκηση 1.2.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 34 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Στις αναπαραστάσεις προσημασμένων αριθμών, το πιο σημαντικό bit δηλώνει το πρόσημο. Πχ ο $(011)_2$ είναι ο $+(3)_{10}$ ενώ ο $(111)_2$ είναι ο $-(3)_{10}$ σε περίπτωση που χρησιμοποιούμε 3 bit πληροφορίας. Λέγοντας «πιο σημαντικό bit» εννοούμε το αριστερότερο bit. Όταν αυτό είναι μηδέν, το πρόσημο είναι «+», ενώ όταν είναι ένα, το πρόσημο είναι «-».

□

Λύση

Πίσω στην Άσκηση 1.2.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 35 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Δουλεύουμε όπως στο δεκαδικό.

□

Λύση

Πίσω στην Άσκηση **1.3.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 36 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Κάντε ότι θα κάνετε σε αφαίρεση δεκαδικών. Θυμηθείτε απλά ότι εδώ $1 + 1 = 10$.

□

Λύση

Πίσω στην Άσκηση 1.3.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 37 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Βρείτε το δυαδικό αντίστοιχο του 23 και το συμπλήρωμα ως προς 1 του δυαδικού αντίστοιχου του -5 και προσθαίστε.

□

Λύση

Πίσω στην Άσκηση 1.3.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 38 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Βρείτε το δυαδικό αντίστοιχο του 68 και το συμπλήρωμα ως προς 1 του δυαδικού αντίστοιχου του -54 και προσθέστε.

□

Λύση

Πίσω στην Άσκηση 1.3.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 39 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Βρείτε το δυαδικό αντίστοιχο του 68 και το συμπλήρωμα ως προς 2 του δυαδικού αντίστοιχου του -54 και προσθέστε.

□

Λύση

Πίσω στην Άσκηση 1.3.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 40 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Βρείτε το δυαδικό αντίστοιχο του 18 και το συμπλήρωμα ως προς 2 του δυαδικού αντίστοιχου του -115 και προσθέστε.

□

Λύση

Πίσω στην Άσκηση 1.3.8



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 41 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

$$1. (1010.001)_2 = 2^3 + 2^1 + 2^{-2} + 2^{-3} = (10.375)_{10}$$

$$2. (630.4)_8 = 6 \times 8^2 + 3 \times 8 + 4 \times 8^{-1} = (408.5)_{10}$$

$$3. (11101)_2 = [\underbrace{(011)}_2 \underbrace{(101)}_2]_8 = (35)_8 \text{ και } (100110101)_2 = [(100)_2(110)_2(101)_2] = (465)_8.$$

$$4. (01011111000)_2 = [\underbrace{(0010)}_2 \underbrace{(1111)}_2 \underbrace{(1000)}_2] = (2F8)_{16}.$$

$$5. (1C5A)_{16} = 1 \times 16^3 + 2 \times 16^2 + 5 \times 16^1 + 10 = (7258)_{10}.$$

□

Πίσω στην Άσκηση 1.1.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 42 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1.

αριθμός $\times 2$	A	K	δυναδικός
$0.625 \times 2 = 1.25$	1	0.25	1
$0.25 \times 2 = 0.5$	0	0.5	10
$0.5 \times 2 = 1.0$	1	0	101

Άρα $(0.625)_{10} = (0.101)_2$.

2. $(0.101011)_2 = 2^{-1} + 2^{-3} + 2^{-5} + 2^{-6} = (0.671875)_{10}$.

□

Πίσω στην Άσκηση 1.1.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 43 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1. Είναι αντίστοιχα 0110 και 0111.
2. Είναι αντίστοιχα 10110 και 10111.

□

Πίσω στην Άσκηση 1.2.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 44 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Το συμπλήρωμα του 001010 ως προς 1 είναι 110101. Άρα το συμπλήρωμα ως προς 1 είναι το 1110101.

Το συμπλήρωμα του 001010 ως προς 2 είναι 110110. Άρα το συμπλήρωμα ως προς 1 είναι το 1110110. □

Πίσω στην Άσκηση 1.2.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 45 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1. 5 bit: $+1011 = 01011$, $-1011 = 11011$ και $\overline{+1011} = 01011$, $\overline{-1011} = 10100$.

2. 7 bit: $+1011 = 0001011$, $-1011 = 1001011$ και $\overline{+1011} = 0110100$, $\overline{-1011} = 1110100$.

□

Πίσω στην Άσκηση 1.2.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 46 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1. $0 + 0 = 0$, $C = 0$
2. $1 + 0 = 1$, $C = 0$
3. $0 + 1 = 1$, $C = 0$
4. $1 + 1 = 10$, $C = 1$.

□

Πίσω στην Άσκηση **1.3.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 47 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

$$\begin{array}{rcccccc} & 1 & 1 & 1 & 1 & & \leftarrow \text{κρατούμενα} \\ 1 & 0 & 1 & 1 & 0 & 1 & \\ + & & 1 & 1 & 1 & 1 & \\ \hline 1 & 1 & 1 & 1 & 0 & 0 & \end{array}$$

□

Πίσω στην Άσκηση 1.3.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 48 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1.

$$\begin{array}{r} 1 \ 0 \ 1 \ 1 \ 1 \\ - \quad 1 \ 0 \ 1 \\ \hline 1 \ 0 \ 0 \ 1 \ 0 \end{array}$$

2.

$$\begin{array}{r} \ 1 \ 1 \ 1 \\ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \\ - \quad 1 \ 1 \ 1 \ 1 \\ \hline 1 \ 1 \ 1 \ 1 \ 0 \end{array} \quad \leftarrow \text{κρατούμενα}$$

□

Πίσω στην Άσκηση 1.3.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 49 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Είναι $(23)_{10} = (10111)_2$ και $(4)_{10} = (100)_2$. Χρησιμοποιώντας 8 bit: $(23)_{10} = (00010111)_2$ και $(4)_{10} = (10000100)_2$ (το πιο σημαντικό bit εδώ δηλώνει το πρόσημο). Το συμπλήρωμα ως προς 2 είναι $\overline{10000100} = 11111011$. Έτσι

$$\begin{array}{r} 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1 \\ +\ 1\ 1\ 1\ 1\ 1\ 0\ 1\ 1 \\ \hline 1\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \end{array} \quad \leftarrow \text{κρατούμενα}$$

Διαγράφουμε το πιο σημαντικό ψηφίο του 1000010010 (διότι έχει ένα επιπλέον bit). Παίρνουμε 000010010 και στη συνέχεια του προσθέτουμε το 1, δηλαδή παίρνουμε τον 10011 που είναι το αποτέλεσμα της ζητούμενης αφαίρεσης. (Ναι! Του προσθέτουμε 1!) □

Πίσω στην Άσκηση 1.3.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 50 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $68 = 01000100$. Ακόμα $-54 = 10110110$ και $\overline{10110110} = 11001001$. Οπότε

$$\begin{array}{r} 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0 \\ +\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 1 \\ \hline 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \end{array}$$

Διαγράφουμε από τον 100001101 το πιο σημαντικό bit και παίρνουμε τον 00001101 στον οποίο προσθέτουμε το 1 : το αποτέλεσμα είναι ο 00001110. \square

Πίσω στην Άσκηση 1.3.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 51 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $68 = 01000100$. Ακόμα $-54 = 10110110$ και $\overline{10110110} = 11001010$. Οπότε

$$\begin{array}{r} 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0 \\ +\ 1\ 1\ 0\ 0\ 1\ 0\ 1\ 0 \\ \hline 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0 \end{array}$$

Αφαιρούμε το πιο σημαντικό ψηφίο από τον 100001110 (διότι πλεονάζει) και παίρνουμε τον 00001110 που είναι το ζητούμενο αποτέλεσμα. \square

Πίσω στην Άσκηση 1.3.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 52 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $18 = 00010010$ και $\overline{-115} = 10001100$.

$$\begin{array}{r} 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \\ +\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 0 \\ \hline 1\ 0\ 0\ 1\ 1\ 1\ 1\ 0 \end{array}$$

Εδώ δεν υπάρχει πλεονάζον ψηφίο. Το αποτέλεσμα της αφαίρεσης είναι 10011110. □

Πίσω στην Άσκηση 1.3.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 53 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $18 = 00010010$ και $\overline{-115} = 10001101$.

$$\begin{array}{r} 0\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \\ +\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 1\ 1\ 1\ 1\ 1 \end{array}$$

Εδώ δεν υπάρχει πλεονάζον ψηφίο. Το αποτέλεσμα της αφαίρεσης είναι 10011111. □

Πίσω στην Άσκηση 1.3.8



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 54 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: $\bar{a} + a = 1$, $a + bc = (a + b)(a + c)$.

□

Λύση

Πίσω στην Άσκηση **2.1.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 55 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Θεωρούμε τις γραμμές εκείνες που έχουν 1 στην τελευταία στήλη. Σε κάθε τέτοια γραμμή αντιστοιχούμε τον κατάλληλο ελαχιστόρο. Πχ στην πρώτη γραμμή αντιστοιχούμε το $\bar{x}_1 \bar{x}_2 \bar{x}_3$, στην πέμπτη το $x_1 \bar{x}_2 \bar{x}_3$ κτλ (δηλαδή, όπου βλέπουμε 0 βάζουμε \bar{x}_i (?=ο αντίστοιχος δείκτης) και όπου 1 βάζουμε x_i .) □

Λύση

Πίσω στην Άσκηση 2.1.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 56 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Karnaugh



Λύση

Πίσω στην Άσκηση **2.1.6**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 57 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Karnaugh



Λύση

Πίσω στην Άσκηση **2.1.7**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 58 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: $0 \equiv 0000 \equiv \bar{x}\bar{y}\bar{z}\bar{w}, 1 \equiv 0001 \equiv \bar{x}\bar{y}\bar{z}w, 2 \equiv 0010 \equiv \bar{x}\bar{y}z\bar{w}, 4 \equiv 0100 \equiv \bar{x}yz\bar{w}, \dots$ □

Λύση

Πίσω στην Άσκηση **2.1.8**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 59 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Κατασκευάστε τον πίνακα αλήθεια των S, C και στη συνέχεια σχεδιάστε το αντίστοιχο κύκλωμα. □

Λύση

Πίσω στην Άσκηση 2.1.9



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 60 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Κατασκευάστε τον πίνακα αλήθειας.



Λύση

Πίσω στην Άσκηση **2.1.10**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 61 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Είναι

$$\begin{aligned}f &= \bar{x}_1 \bar{x}_2 + \bar{x}_1 x_2 + x_1 x_2 \\ &= \bar{x}_1 (\bar{x}_2 + x_2) + x_1 x_2 \\ &= \bar{x}_1 + x_1 x_2 \\ &= (\bar{x}_1 + x_1)(\bar{x}_1 + x_2) \\ &= \bar{x}_1 + x_2 \\ &= g.\end{aligned}$$

□

Πίσω στην Άσκηση **2.1.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 62 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

x_1	x_2	$f(x_1, x_2)$
0	0	1
0	1	1
1	0	0
1	1	1

□

Πίσω στην Άσκηση **2.1.2**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 63 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

□

Πίσω στην Άσκηση **2.1.3**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 64 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $f = \bar{x}_1\bar{x}_2\bar{x}_3 + x_1\bar{x}_2\bar{x}_3 + x_1x_2\bar{x}_3$.

□

Πίσω στην Άσκηση **2.1.4**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 65 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: $f = x_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 x_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3 + x_1 x_2 x_3$.

□

Πίσω στην Άσκηση **2.1.5**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 66 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Ο πίνακας Karnaugh της f είναι ο

x^y	0	1
0	0	1
1	1	1

Άρα $f = x + y$.

Ο πίνακας Karnaugh της g είναι ο

x^y	0	1
0	1	1
1	1	0

Άρα $f = \bar{x} + \bar{y}$.

□

[Πίσω στην Άσκηση 2.1.6](#)



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 67 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Ο πίνακας Karnaugh είναι

$x \backslash yz$	00	01	11	10
0	0	0	1	0
1	1	0	1	1

Άρα $f = yz + x\bar{z}$

□

Πίσω στην Άσκηση **2.1.7**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 68 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Ο πίνακας Karnaugh είναι

$\begin{matrix} zw \\ xy \end{matrix}$	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	1	1	0	1
10	1	1	0	0

Άρα η συνάσταση ελαχιστοποιείται στην

$$f = \bar{z} + \bar{x}\bar{w} + y\bar{w}.$$

□

Πίσω στην Άσκηση **2.1.8**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 69 από 187

Πίσω

Όλη η οθόνη

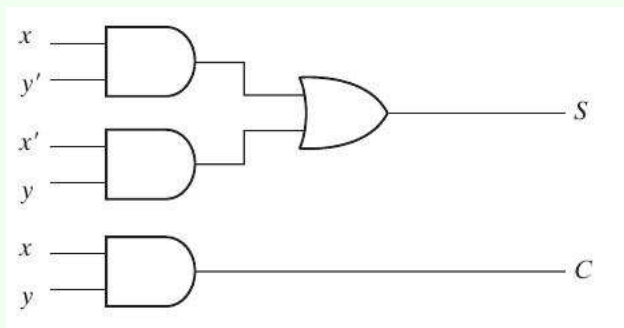
Κλείσε

Έξοδος

Απόδειξη: Ο πίνακας αλήθειας των S, C είναι

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Συνεπώς $C = xy$, $S = x\bar{y} + y\bar{x}$. Τα λογικά κυκλώματα είναι τα



□

Πίσω στην Άσκηση 2.1.9



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 70 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Έστω x, y, z οι είσοδοι. Τότε ο πίνακας αλήθειας είναι ο

x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
1	0	1	1	0
0	1	1	1	0
1	1	1	1	1

και άρα

$$C = xy\bar{z} + x\bar{y}z + \bar{x}yz + xyz = xy + xz + yz, \quad S = \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz.$$

□

Πίσω στην Άσκηση 2.1.10



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 71 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

αλγόριθμος μέσος-όρος

δεδομένα

α , άθροισμα :: πραγματικός

δείκτης :: ακέραιος

αρχή

μετρητής:=0

άθροισμα:=0.0

τύπωσε (' Δώσε ακέραιο (0 για τερματισμό)')

διάβασε (α)

όσο ($\alpha \neq 0$) κάνε

 άθροισμα:=άθροισμα + α

 μετρητής:= μετρητής +1

 τύπωσε (' Δώσε επόμενο αριθμό')

 διάβασε (α)

όσο-τέλος

αν (μετρητής $\neq 0$) τότε

 τύπωσε (' μέσος όρος =', άθροισμα/μετρητής)

διαφορετικά

 τύπωσε (' Δεν δόθηκαν δεδομένα')

αν-τέλος

τέλος

□

Πίσω στην Άσκηση 3.1.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 72 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

αλγόριθμος εναλλαγή

δεδομένα

α: ακέραιος

αρχή

όσο (α μεγαλύτερος από 9 και μικρότερος του 100)

τύπωσε (' Δώσε θετικό διψήφιο')

διάβασε (α)

όσο-τέλος

α:= (πηλίκο του α με το 10) + 10* (υπόλοιπο του α με το 10)

τύπωσε (α)

τέλος

□

Πίσω στην Άσκηση 3.1.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 73 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

αλγόριθμος φυμαλίδα

δεδομένα

i, k, χ : ακέραιος

α : πραγματικό διάνυσμα μήκους 10

αρχή

από $i=1$ έως 10 κάνε

διάβασε το i -οστό στοιχείο του πίνακα

τέλος-κάνε

από $i=1$ έως 10 κάνε

από $k=1$ έως 9 κάνε

αν $\alpha(i)\alpha(k)$ τότε

$\chi = \alpha(i)$

$\alpha(i) = \alpha(k+1)$

$\alpha(k+1) = \chi$

τέλος-αν

τέλος-κάνε

τέλος-κάνε

από $i=1, 10$ κάνε

τύπωσε $\alpha(i)$

τέλος-κάνε

τέλος

□

Πίσω στην Άσκηση 3.1.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 74 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Η συνάρτηση $\text{mod}(x, y)$ υπολογίζει το υπόλοιπο της διαίρεσης του x με το y .

□

Λύση

Πίσω στην Άσκηση **4.1.6**



Συστήματα αρίθμησης

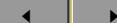
Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 75 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Εισάγουμε μία μεταβλητή, την $temp$. Θέτουμε $temp = x, x = y, y = x$. □

Λύση

Πίσω στην Άσκηση **4.1.7**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 76 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Το $4/3$ στη Fortran είναι μηδέν.

□

Λύση

Πίσω στην Άσκηση **4.2.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 77 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Θέλουμε να συγκρίνουμε τους αριθμούς x, y . Εισάγουμε μία καινούρια μεταβλητή, την \max στην οποία αποθηκεύουμε τη μεγαλύτερη τιμή από τις x, y : Αν $x \geq y$ τότε θέτουμε $\max = x$ διαφορετικά θέτουμε $\max = y$. \square

Λύση

Πίσω στην Άσκηση 4.2.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 78 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Θέλουμε να συγκρίνουμε τους αριθμούς a, b . Εισάγουμε μία καινούρια μεταβλητή, την \min στην οποία αποθηκεύουμε την μικρότερη τιμή από τις a, b : Αν $a \leq b$ τότε θέτουμε $\min = a$ διαφορετικά θέτουμε $\min = b$. \square

Λύση

Πίσω στην Άσκηση 4.2.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 79 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Αν $x \geq 0$ τότε $|x| = x$, διαφορετικά $|x| = -x$.

□

Λύση

Πίσω στην Άσκηση **4.2.5**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 80 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Η εξίσωση $Ax^2 + Bx + C = 0$ έχει λύση, αν η διακρίνουσα $d = B^2 - 4AC$ είναι μη αρνητική. Αν $d > 0$ τότε έχει τις ρίζες $\frac{-B \pm \sqrt{d}}{2A}$, διαφορετικά αν $d = 0$ τότε έχει μία διπλή ρίζα, την $\frac{-B}{2A}$, διαφορετικά είναι αδύνατη. \square

Λύση

Πίσω στην Άσκηση 4.2.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 81 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Δίνεται ότι $C = (F - 32)/1.8$ και $F = 1.8 * C + 32$.

□

Λύση

Πίσω στην Άσκηση **4.2.9**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 82 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: $0! = 1$, $1! = 1$, $n! = 1 \times 2 \times \cdots \times n$, για $n \geq 2$.

□

Λύση

Πίσω στην Άσκηση **4.2.10**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 83 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Τρέξτε το πρόγραμμα.



Λύση

Πίσω στην Άσκηση **4.2.13**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 84 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Ένας ακέραιος μετατρέπεται σε δυαδικό με διαδοχικές διαιρέσεις με το 2. □

Λύση

Πίσω στην Άσκηση **4.2.19**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 85 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Ο αριθμός 123 έχει τρία ψηφία. Η διαίρεση $123/10^3$ δίνει μηδέν στη Fortran. Το πηλίκο ενός τετραψήφιου αριθμού με το 10^4 είναι επίσης μηδέν, αλλά δεν είναι μηδέν το πηλίκο του με το 10^3 . □

Λύση

Πίσω στην Άσκηση 4.2.20



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 86 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Χρησιμοποιείστε ένα do loop για να διαβάσετε τους αριθμούς καθώς εισάγονται, για να υπολογίσετε το άθροισμά τους αλλά και το πλήθος τους. □

Λύση

Πίσω στην Άσκηση 4.3.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 87 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: $\text{Τχνος } (A) = A_{1,1} + A_{2,2} + A_{3,3} + A_{4,4}$.

□

Λύση

Πίσω στην Άσκηση **4.4.4**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 88 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Ανατρέξτε στις ασκήσεις 4.2.2 και 4.2.3.



Λύση

Πίσω στην Άσκηση 4.4.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 89 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Η $f : \mathbb{Z} \rightarrow \mathbb{R}, x \mapsto f(x)$ μετατρέπεται στην real function $f(x)$.

□

Λύση

Πίσω στην Άσκηση **4.5.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 90 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Δες τις ασκήσεις 4.2.10 και 4.5.10.



Λύση

Πίσω στην Άσκηση 4.5.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 91 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Ένας φυσικός n είναι άρτιος αν και μόνο αν το ηλίκο της διαίρεσής του με το 2 είναι 0. Το ηλίκο της διαίρεσης του n με το m δηλώνεται στην Fortran ως $n \text{ mod } (n, m)$. □

Λύση

Πίσω στην Άσκηση 4.5.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 92 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Ο n , $n \geq 1$ είναι πρώτος, αν και μόνο αν δεν διαιρείται από τους $2, 3, \dots, n-1$.
Ο $n \leq -1$ είναι πρώτος, αν και μόνο αν ο $-n$ είναι πρώτος. \square

Λύση

Πίσω στην Άσκηση 4.5.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 93 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Θα πρέπει να μετατρέψετε τις μοίρες σε rad.



Λύση

Πίσω στην Άσκηση **4.5.7**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 94 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Βασιστείτε στο γεγονός ότι $n! = (n - 1)! \times n$.

□

Λύση

Πίσω στην Άσκηση **4.5.10**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 95 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Το πολυώνυμο $a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ μπορεί να παρασταθεί ως ένας πίνακας $(a_0, a_1, a_2, \dots, a_n)$, $n + 1$ στοιχείων. Η παράγωγός του τότε παριστάνεται από τον n -διάστατο πίνακα $(a_1, 2a_2, 3a_3, \dots, na_n)$.

Δημιουργήστε τη συνάρτηση `polynomial_evaluation(a,x,n)` που θα επιστρέφει την τιμή του πολυωνύμου a στο x . Το n είναι ο βαθμός του πολυωνύμου.

Δημιουργήστε την υπορουτίνα `dif_poly(a,b,n)` η οποία θα παραγωγίζει το n -βάθμιο πολυώνυμο a . Το b θα είναι η «επιτρεφόμενη» τιμή της υπορουτίνας, δηλαδή, η παράγωγος του a . □

Λύση

Πίσω στην Άσκηση 4.6.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 96 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Τα μήκη σχηματίζουν τρίγωνο αν το διπλάσιο του μήκους της μεγαλύτερης πλευράς είναι μικρότερο του αθροίσματος των μηκών των τριών πλευρών. (Κάντε χρήση της συνάρτησης `max` της FORTRAN.) □

Λύση

Πίσω στην Άσκηση 4.6.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 97 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Η ορίζουσα τα 2×2 πίνακα $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ είναι ο αριθμός $\det A = ad - bc$. Αν

$\det A \neq 0$ τότε ο πίνακας A έχει αντίστροφο τον $A^{-1} = \frac{1}{\det A} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$ □

Λύση

Πίσω στην Άσκηση 4.6.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 98 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Έστω a, b ακέραιοι με $a \geq b$. Έστω c το υπόλοιπο της διαίρεσης του a με το b (δηλαδή $c = a \bmod b$.) Αν το c είναι μηδέν, τότε ο μέγιστος κοινός διαιρέτης είναι το b . Αν όχι, θέτουμε όπου a το b και b το c και συνεχίζουμε ομοίως. Η διαδικασία τερματίζεται ώσπου να βρούμε υπόλοιπο μηδέν. \square

Λύση

Πίσω στην Άσκηση 4.6.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 99 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη: Γράψτε πρόγραμμα που να ζητά από τον χρήστη την εισαγωγή δύο αριθμών a, b με $f(a)f(b) < 0$.

Γράψτε τη συνάρτηση real function $f()$ η οποία θα ορίζει τη (μαθηματική) συνάρτηση $f(x)$.

Γράψτε τη συνάρτηση real function $\text{solve}(\text{left}, \text{right}, \text{epsilon})$ η οποία θα δέχεται τα πραγματικά ορίσματα $\text{left}, \text{right}, \text{epsilon}$. Τα $\text{left}, \text{right}$ είναι αντίστοιχα το αριστερό και δεξί άκρο του διαστήματος.

Αρχικά θέτουμε $\text{left} = a$ και $\text{right} = b$. Θέτουμε $c = (a+b)/2$ και υπολογίζουμε την τιμή $f(c)$. Αν $f(c) = 0$, έχουμε τελειώσει. Αν όχι, τότε $f(a) \cdot f(c) < 0$, οπότε υπάρχει ρίζα στο διάστημα $[a, c]$, διαφορετικά υπάρχει ρίζα στο $[c, b]$. Στην πρώτη περίπτωση θέτουμε όπου b το c και προχωρούμε ομοίως. Στη δεύτερη περίπτωση θέτουμε όπου a το c και προχωρούμε ομοίως.

Αντιλαμβανόμαστε πως μία τέτοια διαδικασία μπορεί και να μην έχει πεπερασμένα βήματα. Από τη θεωρία γνωρίζουμε ότι με αυτόν τον τρόπο φτιάχνουμε μία ακολουθία αριθμών, που συγκλίνουν στη ρίζα. Τερματίζουμε τη διαδικασία όταν το c γίνει τόσο ώστε $|f(c)| < \text{epsilon}$.

□

Λύση

Πίσω στην Άσκηση 4.6.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 100 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Υπόδειξη:

$$\int_a^b f(x)dx \sim \sum_{i=0}^{n-1} \frac{(f(x_i) + f(x_{i+1})))}{2}$$

όπου $a = x_0 < x_1 < \dots < x_n = b$ διαμέριση του $[a, b]$.

□

Λύση

Πίσω στην Άσκηση 4.6.8

Απόδειξη: 35, 16, 16.25, 0, 0.5



Πίσω στην Άσκηση 4.1.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 101 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 102 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

1. Σωστό
2. Λάθος. Αρχίζει με $_$. Ο πρώτος χαρακτήρας πρέπει να είναι γράμμα.
3. Λάθος. Αρχίζει με αριθμό.
4. Σωστό
5. Λάθος. Δεν επιτρέπεται η $_$.
6. Λάθος. Δεν επιτρέπεται το $_$.

□

Πίσω στην Άσκηση **4.1.2**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 103 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program hello
```

```
  print *, "Hello word!"
```

```
end program hello
```

□

Πίσω στην Άσκηση **4.1.3**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 104 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program first
  integer      :: x
  real         :: y
  double precision :: z

  x=2
  y=5.1
  z=4.52

  print*, "x+y/2.0*sin(z)=", x+y/2.0*sin(z)
end program first
```

□

Πίσω στην Άσκηση 4.1.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 105 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program types
  integer :: i,j
  real    :: x,y

  i = 4.3
  j=1
  x = 5
  y= x/6

  print*, i
  print*, j
  print*, x
  print*, y
end program types
```

□

Πίσω στην Άσκηση 4.1.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 106 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program easter
```

```
integer :: r1, r2, r3, r4, r5, ra, rb, rc  
integer :: year
```

```
print *, "give the year"  
read *, year
```

```
r1 = mod(year, 19)  
r2 = mod(year, 4)  
r3 = mod(year, 7)  
ra = 19 * r1 + 16  
r4 = mod(ra,30)  
rb = 2 * r2 + 4 * r3 + 6 * r4  
r5 = mod(rb, 7)  
rc = r4 + r5
```

```
print *, "easter is ",rc,"days after the 3 of April",year
```

```
end program easter
```

□

Πίσω στην Άσκηση 4.1.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 107 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program swap_program
  real :: x,y
  real :: temp

  print*, "give the numbers x,y"
  read(*,*) x,y
  print*, "before the change x=",x,"y=",y

  temp=x
  x=y
  y=temp
  print*, "after the change x=",x,"y=",y
end program swap_program
```

□

Πίσω στην Άσκηση 4.1.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 108 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program volume
  real :: r ! the sphere's radius

  print*, "Give the shpere's radius"
  read*, r

  if(r>0.) then
    print*, "The volume of the sphere is", 4.0/3.0*3.14*r**3
  else
    print*, "ERROR"
  end if
end program volume
```

□

Πίσω στην Άσκηση 4.2.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 109 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program maximun
  real :: x,y,max
  write(*,*) 'Give me x,y'
  read(*,*) x,y

  if (x .ge. y) then
    max=x
  else
    max=y
  end if

  write(*,*) 'max(' ,x, ', ',y, ')= ',max
end program maximun
```

□

Πίσω στην Άσκηση 4.2.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 110 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program minimum
  REAL :: a, b, min
  PRINT*, "Give a, b"
  READ(*,*) a, b
  IF (a <= b) THEN
    min = a
  ELSE
    min = b
  END IF
  Write(*,*) 'The minimum of ', a, ' and ', &
    b, ' is ', min
end program minimum
```

□

Πίσω στην Άσκηση 4.2.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 111 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program minimum_of_3
  real :: a,b,c
  real :: min

  print*, "Give a,b,c"
  read(*,*) a,b,c

  if (a < b) then
    if (a < c) then
      min = a
    else
      min = c
    end if
  else
    if (b < c) then
      min = b
    else
      min = c
    end if
  end if

  print*, "The minimum is ",min
end program minimum_of_3
```

□

Πίσω στην Άσκηση 4.2.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 112 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program absolute_value
  real :: x,y

  print*, "Give x"
  read(*,*) X

  if (x >= 0) then
    y=x
  else
    y=-x
  end if
  write(*,*) 'The absolute value of', x, ' is ', y
end program absolute_value
```

□

Πίσω στην Άσκηση 4.2.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 113 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program brocker
  real :: x,commission

  print*, "Give the value of the shares"
  read(*,*) x

  if (x<2500.0) then
    commission = 30 + 0.017*x
  else if (x < 6000) then
    commission = 40 + 0.006*x
  else if (x<20000) then
    commission = 60 + 0.002*x
  else if (x<100000) then
    commission = 100 + 0.0011*x
  else
    commission = 150 + 0.0001*x
  end if

  print*, "The commission of",x,"euros is",commission
end program brocker
```

□

Πίσω στην Άσκηση 4.2.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 114 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program QuadraticEquation
```

```
real :: a, b, c
```

```
real :: d ! Discriminant
```

```
real :: root1, root2
```

```
! Read the coefficients a, b, c
```

```
print*, 'Give A, B, C for the equation Ax^2+Bx+C=0: '
```

```
read*, a, b, c
```

```
! Compute the discriminant
```

```
d = sqrt(b*b - 4.0*a*c)
```

```
! Solve the equation
```

```
if (d>0) then
```

```
root1 = (-b + d)/(2.0*a)
```

```
root2 = (-b - d)/(2.0*a)
```

```
print*, "The roots of the equation are ", root1, root2
```

```
else if (d==0) then
```

```
root1 = (-b)/(2.0*a)
```

```
print*, "The equation has a double root, which is ", root1
```

```
else
```

```
print*, "The equation is impossible"
```

```
end if
```

```
end program QuadraticEquation
```

□

Πίσω στην Άσκηση 4.2.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 115 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 116 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program sign_of_integer
```

```
integer          :: x          ! The number  
character(len=1) :: sign      ! The sign
```

```
print*, "Give the number"  
read*, x
```

```
select case (x)  
case ( : -1)          ! x<=-1  
    sign = "-"  
case ( 0 : )         ! x>=0  
    sign = "+"  
end select
```

```
print*, "The sign of the number ",number," is ",sign  
end program sign_of_integer
```

□

Πίσω στην Άσκηση 4.2.8



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 117 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program fareneit_celcius_convertor
character :: c
real      :: x
real      :: celcius , fahrenheit

! we choose what to enter
print*, "Give F for Fareneit, and C for Celcius"
read*, c

print*, "Give the degrees"
read*,  x

select case (c)
case ("C")                ! C->F
    celcius=x
    fahrenheit=9.0/5.0*celcius +32.0
case ("F")
    fahrenheit=x
    celcius=5.0/9.0*(fahrenheit-32.0)    ! F->C
end select

print *, "celcius=",celcius , "hahrenheit=",fahrenheit
end program fareneit_celcius_convertor
```

□

Πίσω στην Άσκηση 4.2.9



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 118 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program factorial
  integer :: n,m,i

  print*,"Give a natural number"
  read(*,*) n

  m=1
  if (n>0) then
    do i=1,n
      m=m*i
    end do
    print*, n,"!=" ,m
  else
    if (n .eq.0) then
      print*, "0!=1"
    end if
  end if
end program factorial
```

□

Πίσω στην Άσκηση 4.2.10



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 119 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program ordering
  real :: a,b,c

  print*, "Give me 3 numbers"
  read*, a,b,c

  if (a < b) then
    ! a < b
    if (a < c) then
      ! a < c : a the smallest
      if (b < c) then
        ! b < c : a < b < c
        print*, a, b, c
      else
        ! c <= b : a < c <= b
        print*, a, c, b
      end if
    else
      ! a >= c : c <= a < b
      print*, c, a, b
    end if
  else
    ! b <= a
    if (b < c) then
      ! b < c : b the smallest
      if (a < c) then
        ! a < c : b <= a < c
        print*, b, a, c
      else
        ! a >= c : b < c <= a
        print*, b, c, a
      end if
    else
      ! c <= b : c <= b <= a
      print*, c, b, a
    end if
  end if
end if
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 120 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

end program ordering



Πίσω στην Άσκηση 4.2.11



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 121 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program grades
  integer          :: x
  character(len=1) :: grade

  print*, "Give the grade"
  read*, x

  if (x < 5) then
    grade = 'f'
  else if (x < 6) then
    grade = 'd'
  else if (x < 7) then
    grade = 'c'
  else if (x < 8) then
    grade = 'b'
  else
    grade = 'a'
  end if

  print*, "the grade is ", grade
end program grades
```

□

Πίσω στην Άσκηση 4.2.12



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 122 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
line:2: minimum
real  :: a, b, minimum
      1
```

Error: Symbol 'minimum' at (1) cannot have a type

Δεν μπορούμε ταυτόχρονα να ονομάζουμε το προγράμμα μας minimum και να ορίζουμε μεταβλητή με το όνομα minimum. □

Πίσω στην Άσκηση 4.2.13



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 123 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program summation
  implicit none
  real      :: sum
  integer   :: i

  sum = 0
  i = 0
  do while (i<=10)
    sum = sum + 3**(i+1)
    i = i+1
  end do

  sum = 0
  i = 0
  do i = 0,10
    sum = sum + 3**(i+1)
  end do

  print*,"The sum is",sum
end program summation
```

□

Πίσω στην Άσκηση 4.2.14



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 124 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program summation
  implicit none
  real      :: sum
  integer   :: i, j

  sum = 0
  i = 0

  ! 1st method. Use the command while
  do while (i<=100)
    j = 0
    do while (j<=200)
      sum = sum + 1.0/float(i+j+1)
      j = j+1
    end do
    i = i+1
  end do

  print*,"The sum is",sum
  print*

  !2nd method
  sum = 0
  do i = 0,100
    do j = 0,200
      sum = sum + 1.0/float(i+j+1)
    end do
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 125 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
end do

print *, "The sum is ", sum
print *
end program summation
```



Πίσω στην Άσκηση **4.2.15**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 126 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program series_sum
  double precision      :: sum, term, epsilon
  integer               :: count

  sum=0.
  term=1.
  epsilon=0.0005

  do while(term >= epsilon)
    sum = sum + term
    count = count + 1
    term = 1.0/(count**2)
  end do

  print *, "1+1/2+...+1/",count,"=",sum
end program series_sum
```

□

Πίσω στην Άσκηση 4.2.16



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 127 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program series
  real    :: sum,k
  integer :: i, fact
  real    :: epsilon =10E-100 ! Precision

  sum=0.
  fact=1
  k=1.
  i=1
  do while(1./fact >= epsilon)
    fact=fact*i
    sum=sum+k/fact
    k=-k
    i=i+1
  end do
  print*, "1-1/2!+1/3!-...+(-1)^",i,"1/",i,"!=", sum
end program series
```

□

Πίσω στην Άσκηση 4.2.17



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 128 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program integer_question
  integer :: n = 5
  integer :: x,i,sum1,sum2,sum3

  sum1=0
  sum2=0
  sum3=0

  do i=1,n
    read*, x
    if(x<=0) then
      sum1=sum1+x**2
    end if
    if(mod(x,2)== 1 .and. mod(x,3)==0) then
      sum2=sum2+x
      if(mod(x,5)==0) then
        sum3=sum3+1
      end if
    end if
  end do
  print*, sum1, sum2, sum3
end program integer_question
```

□

Πίσω στην Άσκηση 4.2.18



Συστήματα αρίθμησης

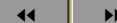
Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 129 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program integer_to_binary
  integer :: n

  print*, "Give an integer"
  read*, n

  print*, "The number is: "
  do while (n > 0)
    print*, mod(n,2)
    n=n/2
  end do
  print*, "(Should be read from the bottom to the top)"

end program integer_to_binary
```

□

Πίσω στην Άσκηση 4.2.19



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 130 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program number_of_digits
  integer :: n, temp, count
  count=0

  print*, "Give a natural number"
  read*, n
  temp=n
  if (n/=0) then
    print*, "The number of digits of 0 is 1"
  else
    do while (temp /=0)
      count=count+1
      temp=temp/10
    end do
    print*, "The number of digits of",n,"is",count
  end if
end program number_of_digits
```

□

Πίσω στην Άσκηση 4.2.20



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 131 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program estimation_sqrt
  real    :: a, sqrt
  integer :: n, i

  print*, "give a positive a and a natural number n"
  read*, a, n

  sqrt=1.0

  do i=0,n
    sqrt=1.0/2.0*(sqrt+ a/sqrt)
  end do

  print*, "an approximation of the square root"
  print*, "of ",a," is ",sqrt
end program estimation_sqrt
```

□

Πίσω στην Άσκηση 4.2.21



Συστήματα αρίθμησης

Κυκλώματα, λογικές συν-
αρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 132 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program histogram
  implicit none

  integer                :: j , i=0
  integer , dimension(4) :: range ! Ta 4 diasthmata
  integer                :: x      ! oi eisagomenoi arijmoï
  integer                :: n      ! to pl'hjoc twn
                                !eisagomenwn arijmwn

  print*, "Dwse to pl'hjoc twn arijmwn pou ja eis'ageic"
  read*, n
  print*, "Dwse ak'eraiouc arijmo'uc"
  do i=1,n
    read*, x
    if(x<=0) then
      range(1) = range(1)+1
    else
      if(x<=10) then
        range(2) = range(2)+1
      else
        if(x<=20) then
          range(3) = range(3)+1
        else
          range(4) = range(4)+1
        end if
      end if
    end if
  end do
end if
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συν-
ναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 133 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
end do
```

```
print *, "To ist'ogramma tw n arijm'wn pou 'edwsec e'inai"  
print *, "(-oo,0] :", ("*", j=1, range(1))  
print *, "(0,10] :", ("*", j=1, range(2))  
print *, "(10,20] :", ("*", j=1, range(3))  
print *, "(20,oo] :", ("*", j=1, range(4))
```

```
end program histogram
```

□

Πίσω στην Άσκηση 4.2.22



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 134 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program test
  integer :: i,j,n

  print*, "Dwse to plh8os twn grammwn:"
  read*, n
  do i=1,n
    print*, (" ",j=1,n-i),(" ",j=1,2*i-1),(" ",j=1,n-i)
  end do
end program test
```

□

Πίσω στην Άσκηση 4.2.23



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 135 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program star_square
  integer :: i,j,n

  print*, "Dwse to plh8os tw n grammwn:"
  read*, n

  print*, ("*", i=1,n)
  do j=1,n-2
    print*, (" ", i=1,n-2), "*"
  end do
  print*, ("*", i=1,n)
end program star_square
```

□

Πίσω στην Άσκηση 4.2.24



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 136 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program integer
```

```
  integer :: i
```

```
  do i = 1,20
```

```
    print *, i
```

```
  end do
```

```
end program integer
```

□

Πίσω στην Άσκηση 4.3.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 137 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program zero
  character :: x

  loop: do
    print*, "Dwse to x"
    read*, x
    if (x=="0") then
      exit loop
    end if
  end do loop

end program zero
```

□

Πίσω στην Άσκηση 4.3.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 138 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program test
  integer :: i,j,n

  print*, "Dwse to plh8os twn grammwn:"
  read*, n

  do i=1,n
    print*, ("*",j=1,i)
  end do
end program test
```

□

Πίσω στην Άσκηση 4.3.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 139 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program mean
  implicit none

  real :: x, s
  integer :: n

  print *, "dwse thetiko arithmo. dwseis arnhtiko gia exodo"

  s = 0.0
  n = 0
  do
    print *, "dwse arithmo"
    read *, x

    if (x < 0.0) exit
    s = s + x
    n = n + 1
  end do

  if (n == 0) then
    print *, "den edwses kanena thetiko!"
  else
    print *, "o mesos oros einai:", s / n
  end if

end program mean
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 140 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Πίσω στην Άσκηση 4.3.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 141 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Έστω $a = (a_1, a_2, \dots, a_n)$ και $b = (b_1, b_2, \dots, b_n)$.

1. $b_i = a_i + 1, i = 1, \dots, n.$

2. $b_i = a_i * 21, i = 1, \dots, n.$

3. $b_i = a_i / c_i, i = 1, \dots, n.$

4. $b_i = 1.0, i = 1, \dots, n.$

5. Είναι λογική έκφραση. Αληθεύει αν $a_i == b_i$ για κάθε i .

6. $b_i = a_i, i = 1, \dots, n.$

7. $b_i = \cos(a_i), i = 1, \dots, n.$

□

Πίσω στην Άσκηση 4.4.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 142 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program inner_product
  real, dimension(3) :: a,b
  integer           :: i
  real              :: inner_prod
  print*, "D'wse to pr'wto di'anusma"
  do i= 1,3
    read*, a(i)
  end do
  print*, "D'wste to de'utero di'anusma"
  do i = 1,3
    read*, b(i)
  end do
  inner_prod = 0.
  do i = 1,3
    inner_prod = inner_prod + a(i)*b(i)
  end do
  print*, "a = (",a(i), i=1,3), ")"
  print*, "b = (",b(i), i=1,3), ")"
  print*, "To eswterik'ο gin'omeno e'ina",inner_prod
end program inner_product
```

□

Πίσω στην Άσκηση 4.4.2



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 143 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program array_sum
  integer :: i           ! i=metrht'hc
  integer, dimension(10) :: a ! di'anusma m'hkouc 10
  integer :: sum

  print*, "D'wse tic suntetagm'enece dian. m'hkouc 10"

  do i=1,10 ! diab'azei m'ia proc m'ia tic suntetagm'enece
    read*, a(i)
  end do

  sum=0
  do i=1,10
    sum =sum+a(i) ! sto t'eloc the diadikas'iac
  end do !to sum ja e'inai to
          ! a(1)+...+a(10)

  print*, "a(1)+...+a(10)=",sum
end program array_sum
```

□

Πίσω στην Άσκηση 4.4.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 144 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program trace_matrix

  integer , dimension(4,4) :: A
  integer :: i,j
  real :: trace

  do i=1,4
    do j=1, 4
      a(i,j) = (i+j)
    end do
  end do

  trace = 0.0
  do i=1,size(a,1)
    trace = trace + a(i,i)
  end do

  print*, "Ο πίνακας A είναι ο"
  do i=1,4
    print*, (A(i,j),j=1,4)
  end do
  print *, "Το ίχνος του A είναι: ", trace

end program trace_matrix
```

□

Πίσω στην Άσκηση **4.4.4**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα **145** από **187**

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 146 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program matrix_multiplication
```

```
integer :: n=2  
real, dimension(2,2) :: A,B,C  
integer :: i,j, k  
real :: sum
```

```
do i=1,n  
do j=1, n  
print*, "d'wse to (",i,j,") stoice'io tou A:"  
read*, A(i,j)  
end do  
end do
```

```
do i=1,n  
do j=1, n  
print*, "d'wse to (",i,j,") stoice'io tou B:"  
read*, B(i,j)  
end do  
end do
```

```
do i=1,n  
do j=1,n  
sum = 0.0  
do k=1,n
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 147 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
        sum = sum + A(i,k) * B(k,j)
    end do
    C(i,j) = sum
end do

print*, "AB="
do i=1,n
    print*, (C(i,j), j=1,n)
end do
end program matrix_multiplication
```

□

Πίσω στην Άσκηση 4.4.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 148 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program min_max
  integer          :: i,max,min
  integer          :: n=10
  integer , dimension(n):: a

  print*, "D'wse ",n," ak'eraiouc"
  do i=1,n
    read*, a(i)
  end do

  max=a(1)
  min=a(1)
  do i=1,n
    if(a(i) >= max) max=a(i)
    if(a(i) <= min) min=a(i)
  end do
  print*, min,max
end program min_max
```

□

Πίσω στην Άσκηση 4.4.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 149 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program integer_to_binary
  integer :: n,temp
  integer, dimension(:), allocatable :: number
  integer :: i=0 ! metrht'hc

  integer :: size ! to pl'hjoc tw'n yhf'iwn
  !the duadik'hc par'astashc
  print*, "D'wse ak'eraio"
  read*, n
  temp=n

  do while (temp > 0) ! metr'ame p'osa yhf'ia 'eqei o (n)_2
    i=i+1
    temp=temp/2
  end do
  size=i

  allocate (number(size))

  temp=n
  do i=1,size ! apojhke'ouome sto array number ta up'oloipa
    number(i) = mod(temp,2)
    temp=temp/2
  end do

  print*, "H diadik'h par'astash tou",n,"e'inai h",&
  (number(i), i=size,1,-1)
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 150 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
end program integer_to_binary
```

□

Πίσω στην Άσκηση 4.4.7



Συστήματα αρίθμησης

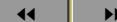
Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 151 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
1. real function f(x)
   implicit none
   real :: x

   f=sin(x**2)
   return
end function f
```

2.

```
program function_demostration

  implicit none
  real :: x,y
  real :: f

  print*, "dwse to x"
  read*, x
  y=f(x)
  print*, "f(",x,")=",y
end program function_demostration

real function f(z)
  implicit none
  real :: z

  f=z**2
```

```
return  
end function f
```

□

Πίσω στην Άσκηση **4.5.1**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 152 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 153 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program function_demostration
  implicit none
  real :: x,y
  real :: f, g

  print*, "dwse x,y"
  read*, x,y

  print*, "f(",x,")=", f(x), "g(",x,y,")=", g(x,y)
end program function_demostration
```

```
real function f(x)
  implicit none
  real :: x

  f=x**2
  return
end function f
```

```
real function g(x,y)
  implicit none
  real ::x,y

  if (x>y) then
    g= x-y
  else
    g=x**2
```

```
end if
return
end function g
```

□

Πίσω στην Άσκηση **4.5.2**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 154 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 155 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program factorial
```

```
  integer :: i  
  integer :: fact
```

```
  print*, (fact(i), i=1,10)
```

```
end program factorial
```

```
integer function fact(n)
```

```
  integer :: n
```

```
  integer :: i
```

```
  fact = 1
```

```
  do i=1,n
```

```
    fact = fact * i
```

```
  end do
```

```
end function fact
```

□

Πίσω στην Άσκηση 4.5.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 156 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program is_even
  integer :: i
  logical :: iseven

  do i=1,10
    if(iseven(i)) then
      print*, i
    end if
  end do
end program is_even

logical function iseven(i)
  if (mod(i,2) == 0) then
    iseven =.TRUE.
  return
  else
    iseven=.FALSE.
  return
  end if
end function iseven
```

□

Πίσω στην Άσκηση 4.5.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 157 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program swap_program
  real :: x,y

  print*, "D'wse touc x,y arijmo'uc"
  read(*,*), x,y
  print*, "Prin thn allag'h x=",x,"y=",y

  call swap(x,y)
  print*, "Met'a thn allag'h x=",x,"y=",y
end program swap_program

subroutine swap(x,y)
  temp=x
  x=y
  y=temp
  return
end subroutine swap
```

□

Πίσω στην Άσκηση 4.5.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 158 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program prime
  integer :: n,i
  integer :: isprime

  print*, "Dwse akeraio n:"
  read*, n

  if (n<=-1) then
    n=-n
  end if

  if (isprime(n)==0) then
    print*, "Den e'inai pr'wtoc."
  else
    print*, "E'inai pr'wtoc."
  end if
end program prime

integer function isprime(n)
  integer :: n,i

  if (n==1 .or. n==0) then
    ret=0
    return
  end if

  do i=2,n-1
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 159 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
if (mod(n, i) == 0) then
  isprime=0
  return
end if
end do
isprime=1
return
end function isprime
```

□

Πίσω στην Άσκηση 4.5.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 160 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program sin_computation
  real sum,k,x
  integer i,fact
  x=35*atan(1.)/45
  sum=0
  k=1.
  i=1
  do i=1,12,2
    sum=sum+k*x**i/fact(i)
    k=-k
  end do
  print*, "H dik'h mac pros'eggish      =",sum
  print*, "H pros'eggish tou sust'hmatoc=",sin(x)
  print*, "Sfalma                        =",abs(sum-sin(x))
end program sin_computation

integer function fact(n)
  integer j,n
  fact=1
  do j=1,n
    fact=fact*j
  end do
  return
end function fact
```

□

Πίσω στην Άσκηση **4.5.7**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα **161** από **187**

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 162 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη: Μία εσωτερική (internal) συνάρτηση δεν μπορεί να περιέχει άλλη συνάρτηση.

□

Πίσω στην Άσκηση 4.5.8



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 163 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program reverse_array
  integer :: n,i
  integer, dimension(:), allocatable :: a,b

  print*, "d'wse to m'hkoc tou dian'usmatoc"
  read*, n

  allocate(a(n))
  allocate(b(n))

  print*, "d'wse to di'anusma a"
  read*, a

  call reverse_ar(a,b,n)
  print*, "H antistrof'h tou",a," e'inai ", b
end program reverse_array

subroutine reverse_ar(a,b,n)
  integer ::n,i
  integer, dimension(n) :: a,b

  do i=1,n
    b(i)=a(n-i+1)
  end do
end subroutine reverse_ar
```

□

Πίσω στην Άσκηση **4.5.9**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 164 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 165 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program factorial_recursive
  integer :: n, factorial

  print*, "D'wse n fusik'ο"
  read*, n
  print*, "n!=", factorial(n)

end program factorial_recursive

integer recursive function factorial(n) result(fact)
  integer :: n
  if (n==1 .or. n==0) then
    fact = 1
  else
    fact =n*factorial(n-1)
  end if
end function factorial
```

□

Πίσω στην Άσκηση 4.5.10



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 166 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program arrays_maximum
  real ,dimension (:), allocatable :: a,b,c
  integer :: n,i

  print*, "Dwse ta mhkh tw n dianysmatwn "
  read*, n

  allocate (a(n))
  allocate (b(n))
  allocate (c(n))

  print*, "Dwse to 1o dianysma, enter gia 'exodo"
  read*, a
  print*, "Dwse to 2o dianysma, enter gia 'exodo"
  read*, b
  call maximum(a,b,c,n)

  print*, "To 'maximum' tw n dianusmatwn einai to ",c
end program arrays_maximum

subroutine maximum(a,b,c,n)
  integer :: n,i
  real ,dimension(n) :: a,b,c

  if (size(a)==size(b)) then
    n=size(a)
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 167 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
do i=1,n
  if (a(i)<=b(i)) then
    c(i)=b(i)
  else
    c(i)=a(i)
  end if
end do
else
  print*, "ERROR"
end if

end subroutine maximum
```

□

Πίσω στην Άσκηση 4.6.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 168 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program polynomial_differentiation
  integer          :: n,i
  real, dimension(:), allocatable :: a,b
  ! a=polyoymo eisodou, b=h paragwgos tou a

  real             :: x ! Ja upolog'isoume to a sto x
  real             :: polynomial_evaluation

  print*, "dwse ton ba8mo toy polywnymou"
  read*, n

  allocate(a(n+1))
  allocate(b(n))
  print*, "dwse to polywnymo a" ! opinakas diastashs n+1
  do i=0,n                      ! h ar'ijmhsh xekin'a apo to 0
    print*, "dwse thn syntetagmenh a_",i
    read*, a(i)
  end do

  print*, "Edwses to polywnymo"
  print*, a(0),"+",(a(i),"x^",i,"+",i=1,n-1),a(n),"x^",n

  print*, "dwse timh sthn opoia ja ypologistei to polu'wnumo"
  read*, x

  print*, "h timh tou polywnymou sto",x,
  print*, "einai",polynomial_evaluation(a,x,n)
```




Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

⏪ ⏩

◀ ▶

Σελίδα 169 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

```
print*, "h paragwgos tou einai h"  
call dif_poly(a,b,n)  
if(n==1) then  
    print *, b(0)  
else if(n==2) then  
    print*, b(0),"+",b(1),"x"  
else  
    print*, b(0),"+",(b(i),"x^",i,"+",i=1,n-2),b(n-1),"x^",n-1  
end if  
end program polynomial_differentiation
```

```
real function polynomial_evaluation(a,x,n)  
integer          :: n,i ! n=ba8mos, i=metrhths  
real, dimension(n+1) :: a ! to polywnymo  
  
do i=0,n  
    polynomial_evaluation=polynomial_evaluation + a(i)*x**i  
end do  
return  
end function polynomial_evaluation
```

```
subroutine dif_poly(a,b,n)  
integer          :: n,i  
real, dimension(n+1) :: a ! to polywnymo eisodoy  
real, dimension(n)   :: b ! h paragwgos tou a  
  
do i=0,n-1
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 170 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
b(i)=(i+1)*a(i+1)
end do
end subroutine dif_poly
```

□

Πίσω στην Άσκηση **4.6.2**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 171 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program triagle
  real a,b,c ! a,b,c oi pleures tou trigwnou.
  integer flag ! flag =1 gia isoskeles
                ! flag =2 gia isopleuro
                ! flag =3 gia skalino

  print*, "dwse 3 8etikous pragmatikous. Apoteloun pleures trigwnou??"
  read*, a,b,c

  if(is_triangle(a,b,c,flag)==1) then
    print*, "Oi pleures sxhmatizoyn"
    if(flag==1) then
      print*, "isoskeles trigwno"
    else
      if(flag==2) then
        print*, "isopleuro trigwno"
      else
        print*, "skalino trigwno."
      end if
    end if
  end if
end program triagle

integer function is_triangle(a,b,c,flag)
  real :: a,b,c ! a,b,c oi pleures tou trigwnou.
  integer :: flag ! flag =1 gia isoskeles
                ! flag =2 gia isopleuro
```

! flag =3 για skalino

```
if(2*max(a,b,c)<= a+b+c) then
  is_triangle=1
  if (a==b .or. a==c .or. b==c) then
    flag = 1
    if(a==b .and. b==c) then
      flag =2
      return
    end if
  else
    flag = 3
    return
  end if
else
  is_triangle=0
  return
end if
end function is_triangle
```

□

Πίσω στην Άσκηση 4.6.3



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 172 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 173 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program inverse_matrix
  real , dimension(2,2) :: a,b ! a o pinakas eisagwghs
                                ! b o antistrofos tou a
  real                   :: invrese ! h synarthsh
                                ! thns antistrofhs
  real                   :: det ! orizousa
  print*, "dwse ton 2x2 pinaka a"
  read*, a

  call inverse(a,b,det)
  if(det == 0) then
    print*, "O pinakas den antistrefetai"
  else
    print*, "O antistrofos tou a einai o"
    print*, b(1,1), b(1,2)
    print*, b(2,1), b(2,2)
    print*, "kai h orizousa tou einai h ",det
  end if
end program inverse_matrix

subroutine inverse(a,b,det)
  real                   :: det ! h orizousa tou a
  real ,dimension(2,2)  :: a,b
  det= a(1,1)*a(2,2) - a(1,2)*a(2,1)

  if(det /= 0) then
    b(1,1)=1.0/det*a(2,2)
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 174 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
b(1,2)=-1.0/det*a(1,2)
b(2,1)=-1.0/det*a(2,1)
b(2,2)=1.0/det*a(1,1)
end if
end subroutine inverse
```

□

Πίσω στην Άσκηση 4.6.4



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 175 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program calendar
  implicit none

  integer :: yyyyymmdd, y, m, d

  do
    ! Zht'ame qronolog'ia.
    ! Plhktrologo'ume 0 gia na fugoume
    print*, "D'wse qronolog'ia sth morf'h yyyyymmdd"
    print*, "D'wse 0 gia 'exodo"
    read*,  yyyyymmdd
    if (yyyyymmdd == 0) exit

    call conversion(yyyyymmdd, y, m, d)

    print*, "year = ", y
    print*, "month = ", m
    print*, "day = ", d
  end do
end program calendar

subroutine conversion(number, year, month, day)
  implicit none

  integer :: number, year, month, day

  year = number / 10000
  month = mod(number, 10000) / 100
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 176 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
day = mod(number, 100)
end subroutine conversion
```

□

Πίσω στην Άσκηση 4.6.5



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 177 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program GreatestCommonDivisor
  implicit none

  integer :: a, b
  integer :: gcd
  print*, "d'wse d'uo akera'iouc"
  read*, a, b
  print*, "O MKD twn",a," kai ",b," e'inai o ", gcd(a, b)
end program GreatestCommonDivisor

integer function gcd(x, y)
  implicit none

  integer :: x, y
  integer :: a, b, c

  if (x<=0) then
    x=-x
  end if
  if (y<=0) then
    y=-y
  end if
  a = x      ! O alg'orijmoc doule'uei gia x>=y. An
  b = y      ! x<=y, all'azoume touc r'olouc twn x,y.
  if (a <= b) then
    c = a
    a = b
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 178 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
        b = c
    end if

do
    c = mod(a, b)
    if (c == 0) exit
    a = b
    b = c
end do

gcd = b
end function gcd
```

□

Πίσω στην Άσκηση 4.6.6



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 179 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program bolzano
  implicit none

  real, parameter :: epsilon = 0.000000001
  real             :: left , fleft
  real             :: right, fright
  real             :: root
  real             :: funct
  real             :: solve

  print*, "l'ush the ex'iswshc f(x) = 0"
  print*, "d'wse to di'asthma [left ,right] 'wste"
  print*, "f(left)f(right)<0."

  read*, left , right

  fleft = funct(left)
  fright = funct(right)

  print*,
  if (fleft*fright > 0.0) then
    print*, "ERROR:: f(left)*f(right)>=0!"
  else
    root = solve(left , right , epsilon)
    print*, "M'ia r'iza metax'u twn",left,"kai",right,"e'inai h ", root
  end if
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 180 από 187

Πίσω

Όλη η σθόνη

Κλείσε

Έξοδος

```
end program bolzano
```

```
!  
! real function funct(x)  
! E'inai h sun'arths'h mac.  
! Thn or'izoume me t'etioio tr'opo, 'wste an  
! to epijumo'ume na th metab'aloume e'ukola.  
!
```

```
real function funct(x)  
  implicit none  
  real          :: x  
  
  funct = x**3  
  return  
end function funct
```

```
real function solve(left , right , epsilon)  
  real          :: left , right , epsilon  
  real          :: a, fa , b, fb , c, fc  
  
  a = left          ! arqikopo'ihsh  
  b = right
```

```

fa = funct(a)
fb = funct(b)
if (abs(fa) < epsilon) then      ! an to |f(a)| e'ina mikr'o
    solve = a                    ! tote to a e'ina r'iza.
else if (abs(fb) < epsilon) then ! an to |f(b)| e'ina mikr'o
    solve = b                    ! tote to b e'ina r'iza.
else
do
    c = (a + b)/2.0              ! to meso tou diast'hatoc [a,b]
    fc = funct(c)
    if (abs(fc) < epsilon) then
        solve = c
        exit
    else if (fa*fc < 0.0) then
        b = c
        fb = fc
    else
        a = c
        fa = fc
    end if
end do
end if
end function solve

```

□

Πίσω στην Άσκηση 4.6.7



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 181 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 182 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program trapezium_rule
  implicit none
  real      :: f           ! h sun'arthsh olokl'hrwshe
  integer   :: n           ! pl'hjoc diasthm'atwn diam'erishe
  real      :: a, b       ! to di'asthma [a,b]
  real      :: integral
  real dx
  real x
  integer   :: i           ! metrht'hc

  print*, "D'wse a, b, n"
  read*, a, b, n

  dx = (b - a) / n
  integral = f(a) + f(b)

  do i = 1, n - 1
    x = a + i * dx
    integral = integral + 2.0 * f(x)
  end do

  integral = integral * dx / 2.0
  print *, "integral = ", integral
end program trapezium_rule

real function f( x )           ! h sun'arthsh olokl'hrwshe
  implicit none
```



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 183 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

```
real ::x
f = sin(x)
return
end function f
```

□

Πίσω στην Άσκηση 4.6.8



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 184 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program bubble
  integer :: i,j,temp
  integer, dimension(10) :: a ! Οι εισαγόμενοι ακέραιοι

  print*, "Dwse 10 ak'eraious arijmo'uc"
  do i=1,10
    read*, a(i)
  end do

  do i=1,10
    do j=i,9
      if(a(i) <= a(j+1)) then
        temp=a(i)
        a(i)=a(j+1)
        a(j+1)=temp
      end if
    end do
  end do

  print*, (a(i),",",i=1,10)

end program bubble
```

□

Πίσω στην Άσκηση 4.6.9



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 185 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program student_scores
  character(len=18) :: student_name
  integer :: i, score_1, score_2, score_3
  real :: weighted_score

  open(unit=2, file="student.records", action="read")
  open(unit=3, file="student.scores", action="write")

  do i = 1, 12
    read (2, '(a18,3(i3))') student_name, score_1, score_2, score_3
    weighted_score = 0.3*score_1 + 0.3*score_2 + 0.4*score_3
    write(3, '(a18,1x,f5.1)') student_name, weighted_score
  end do

end program student_scores
```

□

Πίσω στην Άσκηση 4.7.1



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα

◀ ▶

◀ ▶

Σελίδα 186 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος

Απόδειξη:

```
program number_of_numbers
  integer :: readfile

  print*, readfile(1234), readfile(234)

end program number_of_numbers

integer function readfile(k)
  integer :: k
  integer :: n, i, p

  open (unit = 12, file="numbers", action = "read", status = "old")

  read (12, *) n

  readfile = 0

  do i=1,n
    read (12 , *) p
    if (p == k) readfile = readfile + 1
  end do

  close (unit = 12)

end function readfile
```

□

Πίσω στην Άσκηση **4.7.2**



Συστήματα αρίθμησης

Κυκλώματα, λογικές συναρτήσεις, άλγεβρα Boole

Αλγόριθμοι

Fortran

Τμ. Μαθηματικών

Πρώτη Σελίδα



Σελίδα 187 από 187

Πίσω

Όλη η οθόνη

Κλείσε

Έξοδος