

**ΘΕΜΑ Α**

- A1.** Να γράψετε στο τετράδιό σας τον αριθμό καθεμιάς από τις παρακάτω προτάσεις **1** έως **5** και δίπλα τη λέξη **ΣΩΣΤΟ**, αν η πρόταση είναι σωστή, ή τη λέξη **ΛΑΘΟΣ**, αν η πρόταση είναι λανθασμένη.
1. Ο βρόχος ΓΙΑ i ΑΠΟ 0 ΜΕΧΡΙ 0 δεν εκτελείται καμία φορά.
  2. Σε μια δομή επανάληψης μπορεί να εμφανιστούν λογικά λάθη που σχετίζονται με τη συνθήκη επανάληψης ή τερματισμού.
  3. Υπερχείλιση έχουμε όταν ωθήσουμε ένα στοιχείο σε μια ήδη γεμάτη στοίβα.
  4. Σε πίνακες που είναι ταξινομημένοι χρησιμοποιείται υποχρεωτικά η σειριακή μέθοδος αναζήτησης.
  5. Γενικά, σε περιπτώσεις που η επανάληψη θα συμβεί τουλάχιστον μία φορά, είναι προτιμότερη η χρήση της ΜΕΧΡΙΣ\_ΟΤΟΥ.
- Μονάδες 10**
- A2.** α) Να αναφέρετε τις τυπικές επεξεργασίες των πινάκων. (μονάδες 5)  
 β) Να αναφέρετε τους κανόνες που πρέπει να ακολουθούν οι λίστες των παραμέτρων στα υποπρογράμματα. (μονάδες 3)  
 γ) Να αναφέρετε τέσσερις από τις μαθηματικές συναρτήσεις που περιέχονται στη ΓΛΩΣΣΑ. (μονάδες 4)
- Μονάδες 12**
- A3.** α) Μια στοίβα έξι θέσεων, ύστερα από μερικές ωθήσεις και απωθήσεις, έχει την παρακάτω μορφή:

6	
5	8
4	3
3	7
2	5
1	2

← top

- i) Πόσες απωθήσεις πρέπει να γίνουν προκειμένου να αδειάσει η στοίβα; (μονάδες 2)
  - ii) Να αιτιολογήσετε την απάντησή σας. (μονάδες 3)
- β) Μια ουρά έξι θέσεων, ύστερα από μερικές εισαγωγές και εξαγωγές, έχει την παρακάτω μορφή:

1	2	3	4	5	6
2	5	1	3		
		↑	↑		
		<b>front</b>	<b>rear</b>		

- i) Πόσες εξαγωγές πρέπει να γίνουν προκειμένου να αδειάσει η ουρά; (μονάδες 2)
  - ii) Να αιτιολογήσετε την απάντησή σας. (μονάδες 3)
- Μονάδες 10**

- A4.** Δίνεται το παρακάτω τμήμα προγράμματος:

```
i ← A
ΟΣΟ i ≤ M ΕΠΑΝΑΛΑΒΕ
  ΓΡΑΨΕ i
  i ← i + 2
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

- α) Πόσες φορές θα εκτελεστεί η εντολή εξόδου, όταν η μεταβλητή M πάρει ως τιμή καθεμία από τις παρακάτω εκφράσεις;
- i) A+5
  - ii) A-4
  - iii) A+1
- (μονάδες 6)
- β) Να γράψετε μια αντίστοιχη έκφραση που πρέπει να δοθεί ως τιμή στη μεταβλητή M, ώστε η εντολή εξόδου να εκτελεστεί ακριβώς πέντε (5) φορές. (μονάδες 2)

**ΘΕΜΑ Β**

- B1.** Δίνεται το παρακάτω τμήμα προγράμματος:

ΕΠΙΛΕΞΕ Χ  
 ΠΕΡΙΠΤΩΣΗ 7  
     ΓΡΑΨΕ 'Α'  
 ΠΕΡΙΠΤΩΣΗ 11, 13  
     ΓΡΑΨΕ 'Β'  
 ΠΕΡΙΠΤΩΣΗ < 20  
     ΓΡΑΨΕ 'Γ'  
 ΠΕΡΙΠΤΩΣΗ 50..100  
     ΓΡΑΨΕ 'Δ'  
 ΠΕΡΙΠΤΩΣΗ ΑΛΛΙΩΣ  
     ΓΡΑΨΕ 'Ε'  
 ΤΕΛΟΣ\_ΕΠΙΛΟΓΩΝ

Να γράψετε στο τετράδιό σας ισοδύναμο τμήμα προγράμματος το οποίο να χρησιμοποιεί μόνο μία εντολή AN..ΤΟΤΕ..ΑΛΛΙΩΣ\_ΑΝ, χωρίς επιπλέον εμφωλευμένες εντολές επιλογής.  
 (Η λίστα τιμών 50..100 περιλαμβάνει όλες τιμές από το 50 μέχρι και το 100.)

Μονάδες 10

- B2.** Ένας θετικός ακέραιος αριθμός μεγαλύτερος από το ένα (1) είναι πρώτος αν διαιρείται ακριβώς, μόνο με τον εαυτό του και τη μονάδα. Το παρακάτω τμήμα προγράμματος διαβάζει έναν θετικό ακέραιο αριθμό, ελέγχει αν είναι πρώτος ή όχι και εμφανίζει αντίστοιχο μήνυμα. Για το σκοπό αυτό διαβάζει έναν θετικό ακέραιο  $n$  ( $n > 1$ ), τον διαιρεί διαδοχικά με τους αριθμούς 2, 3, 4, ...,  $n-1$ , ελέγχοντας μετά από κάθε διαίρεση αν ο αριθμός  $n$  διαιρείται ακριβώς. Στην περίπτωση που διαιρείται ακριβώς, σταματάει η επαναληπτική διαδικασία και εμφανίζεται το μήνυμα 'Δεν είναι πρώτος αριθμός'. Αν η επαναληπτική διαδικασία των διαιρέσεων τερματιστεί χωρίς ο αριθμός  $n$  να έχει διαιρεθεί ακριβώς από κανέναν αριθμό εμφανίζεται το μήνυμα 'Είναι πρώτος αριθμός'. Ο αλγόριθμος περιέχει πέντε (5) αριθμημένα κενά. Να γράψετε στο τετράδιό σας τους αριθμούς των κενών και δίπλα ό,τι χρειάζεται να συμπληρωθεί, ώστε το τμήμα προγράμματος να λειτουργεί σωστά.

```

ΔΙΑΒΑΣΕ n
ΠΡΩΤΟΣ ← ... (1) ...
i ← ... (2) ...
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΑΝ ... (3) ... = 0 ΤΟΤΕ
    ΠΡΩΤΟΣ ← ... (4) ...
  ΤΕΛΟΣ_ΑΝ
  i ← i + 1
ΜΕΧΡΙΣ_ΟΤΟΥ i > n - 1 Ή ... (5) ...
ΑΝ ΠΡΩΤΟΣ = ΑΛΗΘΗΣ ΤΟΤΕ
  ΓΡΑΨΕ 'Είναι πρώτος αριθμός'
ΑΛΛΙΩΣ
  ΓΡΑΨΕ 'Δεν είναι πρώτος αριθμός'
ΤΕΛΟΣ_ΑΝ
  
```

Μονάδες 10

### ΘΕΜΑ Γ

Ένα πλοίο μεταφέρει δέματα από λιμάνια της Ελλάδας στην Ιταλία. Σε κάθε λιμάνι που καταπλέει για φόρτωση δηλώνει το βάρος που έχει ήδη φορτωμένο, καθώς και το μέγιστο βάρος που μπορεί να μεταφέρει (όριο βάρους). Η διαδικασία φόρτωσης ελέγχεται από αρμόδιο υπάλληλο.

Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο να υποστηρίζει τη διαδικασία φόρτωσης σε ένα λιμάνι.

Το πρόγραμμα:

**Γ1.** Να περιλαμβάνει κατάλληλο τμήμα δηλώσεων. **Μονάδες 2**

**Γ2.** Να διαβάζει:

- το όριο βάρους του πλοίου (μονάδα 1),
- το βάρος δεμάτων που έχει ήδη φορτωμένα, ελέγχοντας ότι η τιμή του είναι μικρότερη από το όριο βάρους, διαφορετικά να το ξαναζητά (μονάδες 2). **Μονάδες 3**

**Γ3.** Για τη διαδικασία φόρτωσης:

- α) • να εμφανίζει το βάρος που μπορεί ακόμα να φορτωθεί στο πλοίο,  
 • να εμφανίζει το μήνυμα: «ΝΑ ΦΟΡΤΩΘΕΙ ΔΕΜΑ; (ΝΑΙ/ΟΧΙ)»,  
 • να διαβάζει την απάντηση του αρμόδιου υπαλλήλου (χωρίς έλεγχο εγκυρότητας). (μονάδες 3)
- β) Αν η απάντηση είναι «ΝΑΙ»
- να διαβάζει το βάρος του δέματος, να ελέγχει ότι δεν παραβιάζεται το όριο βάρους και να επιτρέπει τη φόρτωσή του, διαφορετικά να εμφανίζει το μήνυμα «ΤΟ ΔΕΜΑ ΔΕΝ ΧΩΡΑΕΙ», (μονάδες 2)
  - εφόσον επιτραπεί η φόρτωσή του, να υπολογίζει και να εμφανίζει το κόστος μεταφοράς του κλιμακωτά, με βάση το βάρος του, ως εξής:  
 -τα πρώτα 500 κιλά χρεώνονται 0,5 € / κιλό,  
 -τα επόμενα 1000 κιλά χρεώνονται 0,3 € / κιλό,

-τα υπόλοιπα χρεώνονται 0,1 € / κιλό. (μονάδες 4)

Η παραπάνω διαδικασία φόρτωσης επαναλαμβάνεται μέχρι να δοθεί ως απάντηση από τον αρμόδιο υπάλληλο η λέξη «ΟΧΙ».

(μονάδες 2)  
**Μονάδες 11**

**Γ4.** Μετά το τέλος φόρτωσης να εμφανίζει:

- πόσα από τα δέματα που ελέγχθηκαν δεν φορτώθηκαν λόγω υπέρβασης του ορίου βάρους (μονάδα 1),
- το συνολικό ποσό που εισπράχθηκε (μονάδα 1),
- το πλήθος των δεμάτων που φορτώθηκαν και είχαν βάρος που ξεπερνούσε τα 1000 κιλά (μονάδες 2).

**Μονάδες 4**

#### **ΘΕΜΑ Δ**

Οι Κινητές Ομάδες Υγείας (ΚΟΜΥ) λαμβάνουν δείγματα βιολογικού υλικού προσώπων για έλεγχο μόλυνσης από τον κορωνοϊό Covid-19. Σε μια περιφέρεια δραστηριοποιούνται 20 ΚΟΜΥ. Κάθε ΚΟΜΥ στη διάρκεια μιας μέρας μπορεί να λάβει μέχρι και 100 δείγματα από μια περιοχή της περιφέρειας. Τα δείγματα αυτά ελέγχονται και κάθε αποτέλεσμα χαρακτηρίζεται ως θετικό (Θ) ή αρνητικό (Α) και καταγράφεται σε πληροφοριακό σύστημα. Να αναπτύξετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο:

**Δ1.** Να περιλαμβάνει κατάλληλο τμήμα δηλώσεων.

**Μονάδες 2**

**Δ2. α)** Να διαβάσει τα ονόματα των περιοχών που δραστηριοποιούνται οι ΚΟΜΥ και να τα καταχωρίζει σε πίνακα με όνομα Π[20] (μονάδα 1).

- β)** Για κάθε ΚΟΜΥ να διαβάσει διαδοχικά τα αποτελέσματα των ελέγχων που έχει πραγματοποιήσει και κάθε αποτέλεσμα να το καταχωρίζει ως ένα γράμμα Α ή Θ στην αντίστοιχη θέση του πίνακα ΑΠ[20,100]. Σε περίπτωση που λήφθηκαν λιγότερα από 100 δείγματα, μετά την καταχώριση του αποτελέσματος του τελευταίου δείγματος διαβάζεται αντί αποτελέσματος η λέξη «ΤΕΛΟΣ», η οποία δεν καταχωρίζεται στον πίνακα. Σε αυτή την περίπτωση τερματίζεται η εισαγωγή τιμών για τη συγκεκριμένη ΚΟΜΥ και το πρόγραμμα καταχωρίζει σε όλες τις υπόλοιπες θέσεις της αντίστοιχης γραμμής το γράμμα Χ (μονάδες 5).

**Μονάδες 6**

**Δ3.** Να εμφανίζει το όνομα ή τα ονόματα των περιοχών που βρέθηκαν τα περισσότερα θετικά δείγματα.

**Μονάδες 6**

**Δ4.** Να εμφανίζει τα ονόματα των περιοχών, ταξινομημένα σε φθίνουσα σειρά ως προς το πλήθος των θετικών δειγμάτων που εντοπίστηκαν. Σε περίπτωση που δύο ή περισσότερες περιοχές έχουν το ίδιο πλήθος θετικών δειγμάτων, τα ονόματά τους να εμφανίζονται με αλφαβητική σειρά. Για την ταξινόμηση να καλείται το υποπρόγραμμα ΤΑΞΙΝΟΜΗΣΗ του ερωτήματος Δ5.

**Μονάδες 3**

**Δ5.** Να αναπτύξετε υποπρόγραμμα με όνομα ΤΑΞΙΝΟΜΗΣΗ, που υλοποιεί τη λειτουργία της ταξινόμησης που περιγράφεται στο ερώτημα Δ4.

**Μονάδες 3**

#### Σημειώσεις

- Για την απάντηση των ερωτημάτων Δ3, Δ4 και Δ5 να θεωρήσετε ότι ο πίνακας ΑΠ έχει συμπληρωθεί σωστά.
- Δεν απαιτούνται έλεγχοι εγκυρότητας τιμών.
- Να θεωρήσετε ότι τα ονόματα των περιοχών είναι διαφορετικά μεταξύ τους

#### **ΑΠΑΝΤΗΣΕΙΣ**

#### **Θέμα Α**

**Α1.**

1. ΛΑΘΟΣ
2. ΣΩΣΤΟ
3. ΣΩΣΤΟ
4. ΛΑΘΟΣ
5. ΣΩΣΤΟ

**Α2.** α) σελ. 165-166 παρ. 9.4

β) σελ. 182 παρ. 10.5

γ) σελ. 131 παρ. 7.6

**Α3.** α) i) 3 αποθήσεις

ii) Η απόθεση γίνεται από την κορυφή της στοιβάς, ως εκ τούτου εφόσον ο δείκτης top έχει την τιμή 3, θα χρειαστεί να γίνουν 3 αποθήσεις για να αδειάσει η στοιβά (top=0).

β) i) Δύο εξαγωγές.

ii) Κάθε φορά που γίνεται μια εξαγωγή από μια ουρά, εξάγεται το στοιχείο που δείχνει ο δείκτης front, ως εκ τούτου θα χρειαστούν δυο εξαγωγές ώστε να αδειάσει η ουρά.

- A4. α) i) 3      ii) καμία      iii) 1  
β) A+9 (σωστό και το A+8)

## ΘΕΜΑ Β

### B1.

ΑΝ X=7 ΤΟΤΕ

ΓΡΑΨΕ 'Α'

ΑΛΛΙΩΣ\_ΑΝ X=11 Η X=13 ΤΟΤΕ

ΓΡΑΨΕ 'Β'

ΑΛΛΙΩΣ\_ΑΝ X<20 ΤΟΤΕ

ΓΡΑΨΕ 'Γ'

ΑΛΛΙΩΣ\_ΑΝ X>=50 ΚΑΙ X<=100 ΤΟΤΕ

ΓΡΑΨΕ 'Δ'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Ε'

ΤΕΛΟΣ\_ΑΝ

### B2.

1. ΑΛΗΘΗΣ
2. 2
3.  $n \text{ MOD } i$
4. ΨΕΥΔΗΣ
5. ΠΡΩΤΟΣ=ΨΕΥΔΗΣ

## ΘΕΜΑ Γ

ΠΡΟΓΡΑΜΜΑ θέμαΓ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΠΛ1, ΠΛ2

ΠΡΑΓΜΑΤΙΚΕΣ: ΟΡΙΟ, ΒΑΡΟΣ\_ΔΕΜΑΤΩΝ, ΒΑΡΟΣ, ΕΛΕΥΘΕΡΟ, ΧΡ, ΣΧΡ

ΧΑΡΑΚΤΗΡΕΣ: ΑΠΑΝΤΗΣΗ

ΑΡΧΗ

ΠΛ1 <- 0

ΠΛ2 <- 0

ΣΧΡ <- 0

!ΕΡΩΤΗΜΑ Γ2

ΔΙΑΒΑΣΕ ΟΡΙΟ

ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ ΒΑΡΟΣ\_ΔΕΜΑΤΩΝ

ΜΕΧΡΙΣ\_ΟΤΟΥ ΒΑΡΟΣ\_ΔΕΜΑΤΩΝ >=0 ΚΑΙ ΒΑΡΟΣ\_ΔΕΜΑΤΩΝ < ΟΡΙΟ

!ΕΡΩΤΗΜΑ Γ3(α)

ΕΛΕΥΘΕΡΟ <- ΟΡΙΟ - ΒΑΡΟΣ\_ΔΕΜΑΤΩΝ

ΓΡΑΨΕ ΕΛΕΥΘΕΡΟ

ΓΡΑΨΕ 'ΝΑ ΦΟΡΤΩΘΕΙ ΔΕΜΑ(ΝΑΙ/ΟΧΙ)'

ΔΙΑΒΑΣΕ ΑΠΑΝΤΗΣΗ

ΟΣΟ ΑΠΑΝΤΗΣΗ = 'ΝΑΙ' ΕΠΑΝΑΛΑΒΕ

ΔΙΑΒΑΣΕ ΒΑΡΟΣ

ΑΝ ΒΑΡΟΣ > ΕΛΕΥΘΕΡΟ ΤΟΤΕ

ΓΡΑΨΕ 'ΤΟ ΔΕΜΑ ΔΕ ΧΩΡΑΕΙ'

ΠΛ1 <- ΠΛ1 + 1

ΑΛΛΙΩΣ

ΑΝ ΒΑΡΟΣ <= 500 ΤΟΤΕ

ΧΡ <- 0.5\*ΒΑΡΟΣ

ΑΛΛΙΩΣ\_ΑΝ ΒΑΡΟΣ <= 1500 ΤΟΤΕ

ΧΡ <- 0.5\*500 + 0.3\*(ΒΑΡΟΣ - 500)

ΑΛΛΙΩΣ

ΧΡ <- 0.5\*500 + 0.3\*1000 + 0.1\*(ΒΑΡΟΣ - 1500)

ΤΕΛΟΣ\_ΑΝ

ΑΝ ΒΑΡΟΣ >1000 ΤΟΤΕ

ΠΛ2 <- ΠΛ2 + 1

ΤΕΛΟΣ\_ΑΝ

```

    ΓΡΑΨΕ ΧΡ
    ΣΧΡ <- ΣΧΡ + ΧΡ
    ΕΛΕΥΘΕΡΟ <- ΕΛΕΥΘΕΡΟ - ΒΑΡΟΣ
    ΒΑΡΟΣ_ΔΕΜΑΤΩΝ <- ΒΑΡΟΣ_ΔΕΜΑΤΩΝ + ΒΑΡΟΣ
    ΤΕΛΟΣ_ΑΝ
    ΓΡΑΨΕ 'ΝΑ ΦΟΡΤΩΘΕΙ ΔΕΜΑ(ΝΑΙ/ΟΧΙ) '
    ΔΙΑΒΑΣΕ ΑΠΑΝΤΗΣΗ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    !ΕΡΩΤΗΜΑ Γ4
    ΓΡΑΨΕ ΠΛ1, ΣΧΡ, ΠΛ2
    ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

```

## ΘΕΜΑ Δ

```

    ΠΡΟΓΡΑΜΜΑ ΘέμαΔ
    ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ: i, j, ΠΛΘ[20], θέση, max
    ΧΑΡΑΚΤΗΡΕΣ: ΑΠ[20, 100], π[20]
    ΑΡΧΗ
    !ΕΡΩΤΗΜΑ Δ2
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
    ΔΙΑΒΑΣΕ π[i]
    j <- 1
    θέση <- 0
    ΟΣΟ j <= 100 ΚΑΙ θέση = 0 ΕΠΑΝΑΛΑΒΕ
    ΔΙΑΒΑΣΕ ΑΠ[i, j]
    ΑΝ ΑΠ[i, j] = 'ΤΕΛΟΣ' ΤΟΤΕ
    θέση <- j
    ΑΠ[i, j] <- 'X'
    ΑΛΛΙΩΣ
    j <- j + 1
    ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΙΑ j ΑΠΟ θέση + 1 ΜΕΧΡΙ 100
    ΑΠ[i, j] <- 'X'
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
    ΠΛΘ[i] <- 0
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 100
    ΑΝ ΑΠ[i, j] = 'Θ' ΤΟΤΕ
    ΠΛΘ[i] <- ΠΛΘ[i] + 1
    ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    !ΕΡΩΤΗΜΑ Δ3
    max <- ΠΛΘ[1]
    ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 20
    ΑΝ ΠΛΘ[i] > max ΤΟΤΕ
    max <- ΠΛΘ[i]
    ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
    ΑΝ ΠΛΘ[i] = max ΤΟΤΕ
    ΓΡΑΨΕ π[i]
    ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    !ΕΡΩΤΗΜΑ Δ4
    ΚΑΛΕΣΕ ΤΑΞΙΝΟΜΗΣΗ(π, ΠΛΘ)
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
    ΓΡΑΨΕ π[i]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
    !ΕΡΩΤΗΜΑ Δ5
    ΔΙΑΔΙΚΑΣΙΑ ΤΑΞΙΝΟΜΗΣΗ(π, ΠΛΘ)
    ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ: i, j, ΠΛΘ[20], t
    ΧΑΡΑΚΤΗΡΕΣ: π[20], x
    ΑΡΧΗ
    ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 20
    ΓΙΑ j ΑΠΟ 20 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
    ΑΝ ΠΛΘ[j] > ΠΛΘ[j - 1] ΤΟΤΕ
    t <- ΠΛΘ[j]
    ΠΛΘ[j] <- ΠΛΘ[j - 1]

```

```
πλο[j - 1] <- t
x <- π[j]
π[j] <- π[j - 1]
π[j - 1] <- x
ΑΛΛΙΩΣ_ΑΝ πλο[j] = πλο[j - 1] ΤΟΤΕ
ΑΝ π[j] < π[j - 1] ΤΟΤΕ
x <- π[j]
π[j] <- π[j - 1]
π[j - 1] <- x
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ
```

Επιμέλεια:

Η Ομάδα Καθηγητών Πληροφορικής Ε.Ο. «ΟΡΙΖΟΝΤΕΣ»  
Μαρία Παπαματθαϊάκη – Γιάννης Κοντάκης – Δημήτρης Βουράκης - Δημήτρης Παπαδάκης