

ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΑΠΑΝΤΗΣΕΙΣ ΠΑΝΕΛΛΑΔΙΚΩΝ**ΠΛΗΡΟΦΟΡΙΚΗ - ΕΠΑΝΑΛΗΠΤΙΚΕΣ****Πέμπτη 10/9/2026****Επιμέλεια: ΜΕΡΕΝΤΙΤΗΣ ΝΙΚΟΣ****ΘΕΜΑ Α****A1. Σωστό – Λάθος**

1. ΣΩΣΤΟ
2. ΣΩΣΤΟ
3. ΛΑΘΟΣ — Η παράσταση ισοδυναμεί με $X > \Psi$, επομένως δεν είναι πάντοτε αληθής.
4. ΣΩΣΤΟ — Τα λογικά λάθη δεν εμποδίζουν τη δημιουργία εκτελέσιμου προγράμματος, αλλά βγάζουν λανθασμένα αποτελέσματα.
5. ΛΑΘΟΣ — Αν ένα δέντρο αποτελείται μόνο από τη ρίζα του, τότε η ρίζα είναι ταυτόχρονα και φύλλο.

A2.**α) Πλεονεκτήματα τμηματικού προγραμματισμού**

Ο τμηματικός προγραμματισμός

1. διευκολύνει τη δημιουργία του προγράμματος,
2. μειώνει τα λάθη
3. επιτρέπει την ευκολότερη παρακολούθηση, κατανόηση και συντήρηση του προγράμματος από τρίτους.

β) Τρόποι περιγραφής και αναπαράστασης αλγορίθμου

1. Ελεύθερο κείμενο
2. Φυσική γλώσσα κατά βήματα
3. Διαγραμματικές τεχνικές, όπως το διάγραμμα ροής
4. Κωδικοποίηση, δηλαδή αναπαράσταση σε ψευδογλώσσα ή γλώσσα προγραμματισμού

A3. Δυαδικά δέντρα αναζήτησης

1. ΟΧΙ — Το B βρίσκεται ως δεξί παιδί του Θ, ενώ είναι μικρότερο από αυτό.
2. ΝΑΙ — Όλοι οι αριστεροί απόγονοι είναι μικρότεροι και όλοι οι δεξιοί μεγαλύτεροι από τον αντίστοιχο κόμβο.
3. ΟΧΙ — Ο κόμβος Θ έπρεπε να είναι δεξιά του Η.

A4. Αντιστοίχιση

Στήλη Α	Στήλη Β
1. Κόμβος	β. Περιέχει δεδομένα
2. Κεφαλή	γ. Δείχνει τον πρώτο κόμβο της λίστας
3. NULL	α. Ο δείκτης του τελευταίου κόμβου της λίστας
4. Δείκτης (pointer)	δ. Δείχνει τον επόμενο κόμβο της λίστας

Τελικές αντιστοιχίσεις:

1–β, 2–γ, 3–α, 4–δ

ΘΕΜΑ Β

B1

Το αρχικό τμήμα εμφανίζει τα αθροίσματα μετά την πρόσθεση των αριθμών:

3, 5, 7, ..., 21

Η αντίστοιχη επανάληψη με ΓΙΑ είναι:

1ος Τρόπος	2ος Τρόπος
$S \leftarrow 0$ ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20 ΜΕ_ΒΗΜΑ 2 $S \leftarrow S + (i + 2)$ ΓΡΑΨΕ S ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	$S \leftarrow 0$ ΓΙΑ i ΑΠΟ 3 ΜΕΧΡΙ 21 ΜΕ_ΒΗΜΑ 2 $S \leftarrow S + i$ ΓΡΑΨΕ S ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

(Δε χρειάζεται αλλά οι τιμές που εμφανίζονται είναι:

3, 8, 15, 24, 35, 48, 63, 80, 99, 120)

B2

Το διάγραμμα ροής μετατρέπεται στην ακόλουθη ψευδογλώσσα:

```

Αλγόριθμος Θέμα_B2
Διάβασε N
x ← 1
S ← 0
Αρχή_επανάληψης
  x ← x * 2
  S ← S + 2
Μέχρις_ότου x >= N
Γράψε S
Τέλος Θέμα_B2
  
```

Η επανάληψη εκτελείται τουλάχιστον μία φορά. Συνεχίζεται όσο $x < N$ και ολοκληρώνεται όταν γίνει $x \geq N$. Αντιστρέψαμε δηλαδή τη συνθήκη που είχε το διάγραμμα.

B3

α) Αρχική μορφή της ουράς

Οι αριθμοί έχουν εισαχθεί διαδοχικά στις θέσεις 1 έως 5.

FRONT = 1

REAR = 5

Αρχική ουρά 7 θέσεων

1	2	3	4	5	6	7
1	6	28	496	8128	—	—
FRONT				REAR		

β) Εξαγωγή τριών αριθμών και ώθησή τους στη στοίβα

Από την ουρά εξαγονται, σύμφωνα με τη λειτουργία FIFO, οι αριθμοί:

1, 6, 28

Οι αριθμοί ωθούνται στη στοίβα με την ίδια σειρά:

Στοίβα μετά τις τρεις ωθήσεις

Θέση	Περιεχόμενο	Δείκτης
6	—	
5	—	
4	—	
3	28	TOP
2	6	
1	1	

Επομένως:

TOP = 3

γ) Απώθηση δύο αριθμών από τη στοίβα

Η στοίβα λειτουργεί με τη μέθοδο LIFO. Επομένως, απωθούνται κατά σειρά:

28 και 6

Απομένει στη στοίβα μόνο ο αριθμός 1.

Τελική μορφή της στοίβας

Θέση	Περιεχόμενο	Δείκτης
6	—	
5	—	
4	—	
3	—	
2	—	
1	1	TOP

Η νέα τιμή του δείκτη είναι:

TOP = 1

δ) Τελική μορφή της ουράς

Από την ουρά εξήχθησαν τα τρία πρώτα στοιχεία. Επομένως, παραμένουν οι αριθμοί:

496, 8128

Τελική μορφή της ουράς

1	2	3	4	5	6	7
—	—	—	496	8128	—	—
			FRONT	REAR		

Οι δείκτες έχουν τις τιμές:

FRONT = 4

REAR = 5

ΘΕΜΑ Γ

ΠΡΟΓΡΑΜΜΑ ΘΕΜΑ_Γ
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: επιλογή, πελάτες, πελάτης_max, πλήθος_τεσσάρων

ΠΡΑΓΜΑΤΙΚΕΣ: κόστος, max, εισπράξεις

ΛΟΓΙΚΕΣ: έχει_σάντουιτς, έχει_σουβλάκι, έχει_καφέ, έχει_νερό

ΑΡΧΗ

πελάτες <- 0

κόστος <- 0

max <- -1

πελάτης_max <- 0

εισπράξεις <- 0

πλήθος_τεσσάρων <- 0

έχει_σάντουιτς <- ΨΕΥΔΗΣ

έχει_σουβλάκι <- ΨΕΥΔΗΣ

έχει_καφέ <- ΨΕΥΔΗΣ

έχει_νερό <- ΨΕΥΔΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ '1. Σάντουιτς'

ΓΡΑΨΕ '2. Σουβλάκι'

ΓΡΑΨΕ '3. Καφές'

ΓΡΑΨΕ '4. Νερό'

ΓΡΑΨΕ '5. Ολοκλήρωση παραγγελίας'

ΓΡΑΨΕ '6. Κλείσιμο καντίνας'

ΔΙΑΒΑΣΕ επιλογή

ΑΝ επιλογή = 1 ΤΟΤΕ

κόστος <- κόστος + 5

έχει_σάντουιτς <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 2 ΤΟΤΕ

κόστος <- κόστος + 3

έχει_σουβλάκι <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 3 ΤΟΤΕ

κόστος <- κόστος + 2.5

έχει_καφέ <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 4 ΤΟΤΕ

κόστος <- κόστος + 0.5

έχει_νερό <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 5 ΤΟΤΕ

πελάτες <- πελάτες + 1

ΓΡΑΨΕ 'Κόστος παραγγελίας: ', κόστος, ' ευρώ'

εισπράξεις <- εισπράξεις + κόστος

ΑΝ κόστος > max ΤΟΤΕ

max <- κόστος

πελάτης_max <- πελάτες

ΤΕΛΟΣ_ΑΝ

ΑΝ έχει_σάντουιτς ΚΑΙ έχει_σουβλάκι ΚΑΙ έχει_καφέ ΚΑΙ έχει_νερό ΤΟΤΕ

πλήθος_τεσσάρων <- πλήθος_τεσσάρων + 1

ΤΕΛΟΣ_ΑΝ

! Αρχικοποίηση για την επόμενη παραγγελία

κόστος <- 0

```
έχει_σάντουιτς <- ΨΕΥΔΗΣ  
έχει_σουβλάκι <- ΨΕΥΔΗΣ  
έχει_καφέ <- ΨΕΥΔΗΣ  
έχει_νερό <- ΨΕΥΔΗΣ  
ΤΕΛΟΣ_ΑΝ  
ΜΕΧΡΙΣ_ΟΤΟΥ επιλογή = 6  
ΓΡΑΨΕ max  
ΓΡΑΨΕ πελάτης_max  
ΓΡΑΨΕ 'Συνολικές εισπράξεις ημέρας: ', εισπράξεις, ' ευρώ'  
ΓΡΑΨΕ 'Παραγγελίες που περιείχαν και τα 4 είδη: ', πλήθος_τεσσάρων  
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

ΘΕΜΑ Δ

ΠΡΟΓΡΑΜΜΑ Αλμα_εις_μήκος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i, j, πλήθος_άκυρων[8], θέση_πρώτης

ΠΡΑΓΜΑΤΙΚΕΣ: ΕΠ[8, 6], μέγιστη_επίδοση

ΧΑΡΑΚΤΗΡΕΣ: ΟΝ[8], ΧΩΡ[8]

ΛΟΓΙΚΕΣ: υπάρχει_χωρίς_έγκυρο

ΑΡΧΗ

! Εισαγωγή των στοιχείων και των επιδόσεων

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΔΙΑΒΑΣΕ ΟΝ[i]

ΔΙΑΒΑΣΕ ΧΩΡ[i]

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6

ΔΙΑΒΑΣΕ ΕΠ[i, j]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

! Πλήθος άκυρων αλμάτων κάθε αθλήτριας

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

πλήθος_άκυρων[i] <- ΑΚΥΡΑ_ΑΛΜΑΤΑ(ΕΠ, i)

ΓΡΑΨΕ ΟΝ[i], πλήθος_άκυρων[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

! Έλεγχος αν υπάρχει αθλήτρια χωρίς έγκυρο άλμα

υπάρχει_χωρίς_έγκυρο <- ΨΕΥΔΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΑΝ πλήθος_άκυρων[i] = 6 ΤΟΤΕ

υπάρχει_χωρίς_έγκυρο <- ΑΛΗΘΗΣ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ υπάρχει_χωρίς_έγκυρο ΤΟΤΕ

ΓΡΑΨΕ 'Υπάρχει αθλήτρια χωρίς έγκυρο άλμα'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν υπάρχει αθλήτρια χωρίς έγκυρο άλμα'

ΤΕΛΟΣ_ΑΝ

! Εύρεση της αθλήτριας με την καλύτερη έγκυρη επίδοση

μέγιστη_επίδοση <- -1

θέση_πρώτης <- 0

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6

ΑΝ ΕΠ[i, j] > μέγιστη_επίδοση ΤΟΤΕ

μέγιστη_επίδοση <- ΕΠ[i, j]

θέση_πρώτης <- i

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Χρυσό μετάλλιο'

ΓΡΑΨΕ 'Όνομα αθλήτριας: ', ΟΝ[θέση_πρώτης]

ΓΡΑΨΕ 'Χώρα προέλευσης: ', ΧΩΡ[θέση_πρώτης]

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

```
ΣΥΝΑΡΤΗΣΗ ΑΚΥΡΑ_ΑΛΜΑΤΑ(ΕΠ, γραμμή): ΑΚΕΡΑΙΑ
ΜΕΤΑΒΛΗΤΕΣ
  ΑΚΕΡΑΙΕΣ: γραμμή, j, πλήθος
  ΠΡΑΓΜΑΤΙΚΕΣ: ΕΠ[8, 6]
ΑΡΧΗ
  πλήθος <- 0
  ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6
    ΑΝ ΕΠ[γραμμή, j] = -1 ΤΟΤΕ
      πλήθος <- πλήθος + 1
    ΤΕΛΟΣ_ΑΝ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΑΚΥΡΑ_ΑΛΜΑΤΑ <- πλήθος
ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ
```

ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΑΠΑΝΤΗΣΕΙΣ

ΠΛΗΡΟΦΟΡΙΚΗ – ΕΠΑΝΑΛΗΠΤΙΚΕΣ

ΘΕΜΑ Α



A1 Σωστό – Λάθος

- 1 **ΣΩΣΤΟ** – Στον έλεγχο μαύρου κουτιού εξετάζεται η λειτουργία του προγράμματος με έγκυρες αλλά και μη έγκυρες τιμές εισόδου, χωρίς να εξετάζεται ο εσωτερικός κώδικας.
- 2 **ΣΩΣΤΟ** – Η απόκρυψη των λεπτομερειών υλοποίησης και η συγκέντρωση δεδομένων και μεθόδων μέσα σε μία κλάση ονομάζεται ενθυλάκωση.
- 3 **ΛΑΘΟΣ** – Η παράσταση ισοδυναμεί με $X > \Psi$, επομένως δεν είναι πάντοτε αληθής.
- 4 **ΣΩΣΤΟ** – Τα λογικά λάθη δεν εμποδίζουν τη δημιουργία εκτελέσιμου προγράμματος, αλλά βγάζουν λανθασμένα αποτελέσματα.
- 5 **ΛΑΘΟΣ** – Αν ένα δέντρο αποτελείται μόνο από τη ρίζα του, τότε η ρίζα είναι ταυτόχρονα και φύλλο.

A2

α) Πλεονεκτήματα τμηματικού προγραμματισμού

Ο τμηματικός προγραμματισμός:

- διευκολύνει τη δημιουργία του προγράμματος,
- μειώνει τα λάθη,
- επιτρέπει την ευκολότερη παρακολούθηση, κατανόηση και συντήρηση του προγράμματος από τρίτους.

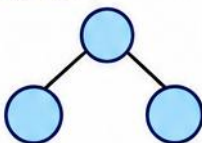
β) Τρόποι περιγραφής και αναπαράστασης αλγορίθμου

- 1 Ελεύθερο κείμενο
- 2 Φυσική γλώσσα κατά βήματα
- 3 Διαγραμματικές τεχνικές, όπως το διάγραμμα ροής
- 4 Κωδικοποίηση, δηλαδή αναπαράσταση σε ψευδογλώσσα ή γλώσσα προγραμματισμού

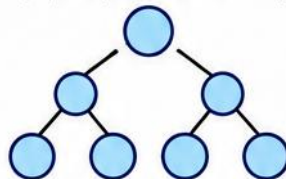


A3 Δυαδικά δέντρα αναζήτησης

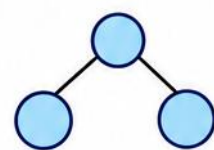
- 1 **ΟΧΙ** – Το B βρίσκεται ως δεξί παιδί του Θ, ενώ είναι μικρότερο από αυτό.



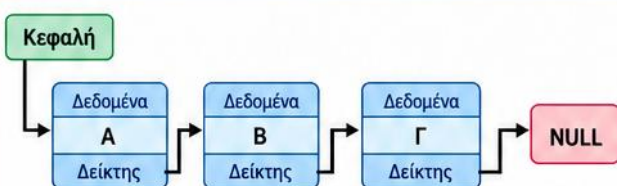
- 2 **ΝΑΙ** – Όλοι οι αριστεροί απόγονοι είναι μικρότεροι και όλοι οι δεξιοί μεγαλύτεροι από τον αντίστοιχο κόμβο.



- 3 **ΟΧΙ** – Ο κόμβος Θ έπρεπε να είναι δεξιά του Η.



A4 Αντιστοίχιση



1.	Κόμβος	β. Περιέχει δεδομένα
2.	Κεφαλή	γ. Δείχνει τον πρώτο κόμβο της λίστας
3.	NULL	α. Ο δείκτης του τελευταίου κόμβου της λίστας
4.	Δείκτης (pointer)	δ. Δείχνει τον επόμενο κόμβο της λίστας

Τελικές αντιστοιχίες:

1-β, 2-γ, 3-α, 4-δ

Επιμέλεια: ΜΕΡΕΝΤΙΤΗΣ ΝΙΚΟΣ



ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΑΠΑΝΤΗΣΕΙΣ

ΠΛΗΡΟΦΟΡΙΚΗ – ΕΠΑΝΑΛΗΠΤΙΚΕΣ

ΘΕΜΑ Β



B1 Το αρχικό τμήμα εμφανίζει τα αθροίσματα μετά την πρόσθεση των αριθμών:
3, 5, 7, ..., 21

Η αντίστοιχη επανάληψη με ΓΙΑ είναι:

1ος Τρόπος

```
S ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20 ΜΕ_ΒΗΜΑ 2
    S ← S + (i + 2)
ΓΡΑΨΕ S
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

2ος Τρόπος

```
S ← 0
ΓΙΑ i ΑΠΟ 3 ΜΕΧΡΙ 21 ΜΕ_ΒΗΜΑ 2
    S ← S + i
ΓΡΑΨΕ S
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

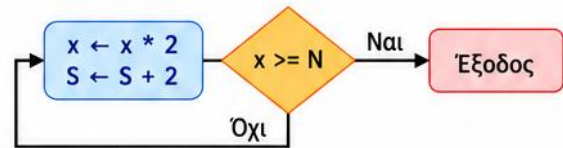
(Δε χρειάζεται αλλά οι τιμές που εμφανίζονται είναι:
3, 8, 15, 24, 35, 48, 63, 80, 99, 120)

B2 Το διάγραμμα ροής μετατρέπεται στην ακόλουθη ψευδογλώσσα:

Αλγόριθμος θέμα_B2

```
Διάβασε N
x ← 1
S ← 0
Αρχή_επανάληψης
    x ← x * 2
    S ← S + 2
Μέχρις_ότου x >= N
Γράψε S
Τέλος θέμα_B2
```

Η επανάληψη εκτελείται τουλάχιστον μία φορά. Συνεχίζεται όσο $x < N$ και ολοκληρώνεται όταν γίνει $x \geq N$. Αντιστρέψαμε δηλαδή τη συνθήκη που είχε το διάγραμμα.



B3

α) Αρχική μορφή της ουράς

Οι αριθμοί έχουν εισαχθεί διαδοχικά στις θέσεις 1 έως 5.

FRONT = 1

Αρχική ουρά 7 θέσεων:

Θέσεις:	1	2	3	4	5	6	7
Περιεχόμενο:	1	6	28	496	8128	—	—
Δείκτες:	↑ FRONT				↑ REAR		

β) Εξαγωγή τριών αριθμών και ώθησή τους στη στοίβα

Από την ουρά εξαγονται, σύμφωνα με τη λειτουργία FIFO, οι αριθμοί: 1, 6, 28

Οι αριθμοί ωθούνται στη στοίβα με την ίδια σειρά:

Στοίβα 6 θέσεων, από πάνω προς τα κάτω:

Θέση:	Περιεχόμενο:
6	—
5	—
4	—
3	28
2	6
1	1

Επομένως:
TOP = 3

γ) Απώθηση δύο αριθμών από τη στοίβα

Η στοίβα λειτουργεί με τη μέθοδο LIFO. Επομένως, απωθούνται κατά σειρά: 28 και 6

Απομένει στη στοίβα μόνο ο αριθμός 1.

Τελική στοίβα 6 θέσεων, από πάνω προς τα κάτω:

Θέση:	Περιεχόμενο:
6	—
5	—
4	—
3	—
2	—
1	1

Η νέα τιμή του δείκτη είναι:
TOP = 1

↑ TOP

δ) Τελική μορφή της ουράς

Από την ουρά εξήχθησαν τα τρία πρώτα στοιχεία. Επομένως, παραμένουν οι αριθμοί: 496, 8128

Τελική ουρά 7 θέσεων:

Θέσεις:	1	2	3	4	5	6	7
Περιεχόμενο:	—	—	—	496	8128	—	—
Δείκτες:				↑ FRONT	↑ REAR		

Οι δείκτες έχουν τις τιμές:

FRONT = 4

REAR = 5

Επιμέλεια: ΜΕΡΕΝΤΙΤΗΣ ΝΙΚΟΣ

ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΑΠΑΝΤΗΣΕΙΣ

ΠΛΗΡΟΦΟΡΙΚΗ – ΕΠΑΝΑΛΗΠΤΙΚΕΣ 2026

ΘΕΜΑ Γ



Πρόγραμμα

ΠΡΟΓΡΑΜΜΑ ΘΕΜΑ_Γ

ΜΕΤΑΒΑΝΤΕΣ

ΑΚΕΡΑΙΕΣ: επιλογή, πελάτες, πελάτης_max, πλήθος_τεσσάρων

ΠΡΑΓΜΑΤΙΚΕΣ: κόστος, max, εισπράξεις

ΛΟΓΙΚΕΣ: έχει_σάντουιτς, έχει_σουβλάκι, έχει_καφέ, έχει_νερό

ΑΡΧΗ

πελάτες <- 0

κόστος <- 0

max <- -1

πελάτης_max <- 0

εισπράξεις <- 0

πλήθος_τεσσάρων <- 0

έχει_σάντουιτς <- ΨΕΥΔΗΣ

έχει_σουβλάκι <- ΨΕΥΔΗΣ

έχει_καφέ <- ΨΕΥΔΗΣ

έχει_νερό <- ΨΕΥΔΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ '1. Σάντουιτς'

ΓΡΑΨΕ '2. Σουβλάκι'

ΓΡΑΨΕ '3. Καφέ'

ΓΡΑΨΕ '4. Νερό'

ΓΡΑΨΕ '5. Ολοκλήρωση παραγγελίας'

ΓΡΑΨΕ '6. Κλείσιμο καντίνας'

ΔΙΑΒΑΣΕ επιλογή

ΑΝ επιλογή = 1 ΤΟΤΕ

κόστος <- κόστος + 5

έχει_σάντουιτς <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 2 ΤΟΤΕ

κόστος <- κόστος + 3

έχει_σουβλάκι <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 3 ΤΟΤΕ

κόστος <- κόστος + 2.5

έχει_καφέ <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 4 ΤΟΤΕ

κόστος <- κόστος + 0.5

έχει_νερό <- ΑΛΗΘΗΣ

ΑΛΛΙΩΣ_ΑΝ επιλογή = 5 ΤΟΤΕ

πελάτες <- πελάτες + 1

ΓΡΑΨΕ 'Κόστος παραγγελίας: ', κόστος, ' ευρώ'

εισπράξεις <- εισπράξεις + κόστος

ΑΝ κόστος > max ΤΟΤΕ

max <- κόστος

πελάτης_max <- πελάτες

ΤΕΛΟΣ_ΑΝ

ΑΝ έχει_σάντουιτς ΚΑΙ έχει_σουβλάκι ΚΑΙ

έχει_καφέ ΚΑΙ έχει_νερό ΤΟΤΕ

πλήθος_τεσσάρων <- πλήθος_τεσσάρων + 1

ΤΕΛΟΣ_ΑΝ

! Αρχικοποίηση για την επόμενη παραγγελία

κόστος <- 0

έχει_σάντουιτς <- ΨΕΥΔΗΣ

έχει_σουβλάκι <- ΨΕΥΔΗΣ

έχει_καφέ <- ΨΕΥΔΗΣ

έχει_νερό <- ΨΕΥΔΗΣ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ επιλογή = 6

ΓΡΑΨΕ max

ΓΡΑΨΕ πελάτης_max

ΓΡΑΨΕ 'Συνολικές εισπράξεις ημέρας: ', εισπράξεις, ' ευρώ'

ΓΡΑΨΕ 'Παραγγελίες που περιείχαν και τα 4 είδη: ', πλήθος_τεσσάρων

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Γ1 Δηλώσεις και αρχικοποίηση

Οι λογικές μεταβλητές καταγράφουν αν η τρέχουσα παραγγελία περιέχει καθένα από τα τέσσερα προϊόντα.

Μεταβλητές	
επιλογή, πελάτες, πελάτης_max, πλήθος_τεσσάρων	(ακέραιες)
κόστος, max, εισπράξεις	(πραγματικές)
έχει_σάντουιτς, έχει_σουβλάκι, έχει_καφέ, έχει_νερό	(λογικές)

Γ2 Καταχώριση παραγγελίας

Κάθε επιλογή προϊόντος αυξάνει το κόστος και ενεργοποιεί την αντίστοιχη λογική μεταβλητή.

1. Σάντουιτς 5 €	2. Σουβλάκι 3 €	3. Καφέ 2.5 €	4. Νερό 0.5 €

Γ3 Μέγιστη παραγγελία

Με την επιλογή 5 αυξάνεται ο αριθμός πελάτη. Αν το κόστος είναι μεγαλύτερο από το max, αποθηκεύονται το νέο μέγιστο κόστος και ο αριθμός του πελάτη.



max
(μέγιστο κόστος)
πελάτης_max
(αριθμός πελάτη)

Γ4 Συνολικές εισπράξεις

Κάθε ολοκληρωμένη παραγγελία προστίθεται στη μεταβλητή εισπράξεις.



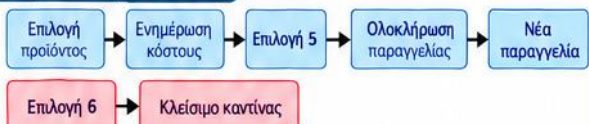
Συνολικές
εισπράξεις
ημέρας
€ € €

Γ5 Παραγγελίες με όλα τα είδη

Αν και οι τέσσερις λογικές μεταβλητές είναι ΑΛΗΘΕΙΣ, αυξάνεται το πλήθος_τεσσάρων. Στη συνέχεια οι μεταβλητές της παραγγελίας μηδενίζονται για τον επόμενο πελάτη.

				πλήθος_τεσσάρων (παραγγελίες)
--	--	--	--	----------------------------------

Ροή εκτέλεσης (συνοπτικά)



Επιμέλεια: ΜΕΡΕΝΤΙΤΗΣ ΝΙΚΟΣ



ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΑΠΑΝΤΗΣΕΙΣ

ΠΛΗΡΟΦΟΡΙΚΗ – ΕΠΑΝΑΛΗΠΤΙΚΕΣ 2026

ΘΕΜΑ Δ



Πρόγραμμα

ΠΡΟΓΡΑΜΜΑ Αλμα_εις_μήκος

ΜΕΤΑΒΑΝΤΕΣ

ΑΚΕΡΑΙΕΣ: i, j , πλήθος_άκυρων[8], θέση_πρώτης

ΠΡΑΓΜΑΤΙΚΕΣ: ΕΠ[8, 6], μέγιστη_επίδοση

ΧΑΡΑΚΤΗΡΕΣ: ΟΝ[8], ΧΩΡ[8]

ΛΟΓΙΚΕΣ: υπάρχει_χωρίς_έγκυρο

ΑΡΧΗ

! Εισαγωγή των στοιχείων και των επιδόσεων

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΔΙΑΒΑΣΕ ΟΝ[i]

ΔΙΑΒΑΣΕ ΧΩΡ[i]

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6

ΔΙΑΒΑΣΕ ΕΠ[i, j]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

! Πλήθος άκυρων αλμάτων κάθε αθλήτριας

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

πλήθος_άκυρων[i] ← ΑΚΥΡΑ_ΑΛΜΑΤΑ(ΕΠ, i)

ΓΡΑΨΕ ΟΝ[i], πλήθος_άκυρων[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

! Έλεγχος αν υπάρχει αθλήτρια χωρίς έγκυρο άλμα

υπάρχει_χωρίς_έγκυρο ← ΨΕΥΔΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΑΝ πλήθος_άκυρων[i] = 6 ΤΟΤΕ

υπάρχει_χωρίς_έγκυρο ← ΑΛΗΘΗΣ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ υπάρχει_χωρίς_έγκυρο ΤΟΤΕ

ΓΡΑΨΕ 'Υπάρχει αθλήτρια χωρίς έγκυρο άλμα'

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν υπάρχει αθλήτρια χωρίς έγκυρο άλμα'

ΤΕΛΟΣ_ΑΝ

! Εύρεση της αθλήτριας με την καλύτερη έγκυρη επίδοση

μέγιστη_επίδοση ← -1

θέση_πρώτης ← 0

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 8

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6

ΑΝ ΕΠ[i, j] > μέγιστη_επίδοση ΤΟΤΕ

μέγιστη_επίδοση ← ΕΠ[i, j]

θέση_πρώτης ← i

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Χρυσό μετάλλιο'

ΓΡΑΨΕ 'Όνομα αθλήτριας: ', ΟΝ[θέση_πρώτης]

ΓΡΑΨΕ 'Χώρα προέλευσης: ', ΧΩΡ[θέση_πρώτης]

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

ΣΥΝΑΡΤΗΣΗ ΑΚΥΡΑ_ΑΛΜΑΤΑ(ΕΠ, γραμμή): ΑΚΕΡΑΙΑ

ΜΕΤΑΒΑΝΤΕΣ

ΑΚΕΡΑΙΕΣ: γραμμή, j , πλήθος

ΠΡΑΓΜΑΤΙΚΕΣ: ΕΠ[8, 6]

ΑΡΧΗ

πλήθος ← 0

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 6

ΑΝ ΕΠ[γραμμή, j] = -1 ΤΟΤΕ

πλήθος ← πλήθος + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΚΥΡΑ_ΑΛΜΑΤΑ ← πλήθος

ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ

Δ1 Πίνακες δεδομένων

Οι πίνακες ΟΝ[8] και ΧΩΡ[8] αποθηκεύουν τα στοιχεία των 8 αθλητριών. Ο διδιάστατος πίνακας ΕΠ[8, 6] αποθηκεύει τα 6 άλματα κάθε αθλήτριας. Το -1 δηλώνει άκυρο άλμα.

ΟΝ[8]	ΧΩΡ[8]	ΕΠ[8, 6]					
		1	2	3	4	5	6
1	1						-1
2	2						
...	...						
8	8						

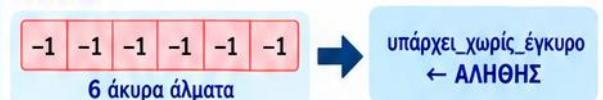
Δ2 Πλήθος άκυρων αλμάτων

Για κάθε αθλήτρια καλείται η συνάρτηση ΑΚΥΡΑ_ΑΛΜΑΤΑ και το αποτέλεσμα αποθηκεύεται στο πλήθος_άκυρων[i].



Δ3 Αθλήτρια χωρίς έγκυρο άλμα

Αν πλήθος_άκυρων[i] = 6, τότε και τα έξι άλματα της αθλήτριας είναι άκυρα και η λογική μεταβλητή γίνεται ΑΛΗΘΗΣ.



Δ4 Χρυσό μετάλλιο

Εξετάζονται όλες οι επιδόσεις. Κάθε τιμή μεγαλύτερη από τη μέγιστη_επίδοση ενημερώνει το μέγιστο και τη θέση_πρώτης. Η αρχική τιμή -1 επιτρέπει να αγνοούνται τα άκυρα άλματα.



μέγιστη_επίδοση
θέση_πρώτης
(αθλήτρια με την καλύτερη
έγκυρη επίδοση)

Δ5 Συνάρτηση

Η ΑΚΥΡΑ_ΑΛΜΑΤΑ δέχεται τον πίνακα ΕΠ και μία γραμμή. Ελέγχει τα 6 στοιχεία της γραμμής, μετρά όσα είναι ίσα με -1 και επιστρέφει το πλήθος.



Ροή εκτέλεσης (συνοπτικά)



Επιμέλεια: ΜΕΡΕΝΤΙΤΗΣ ΝΙΚΟΣ