

ΕΞΕΤΑΣΕΙΣ Γ΄ ΤΑΞΗΣ ΗΜΕΡΗΣΙΟΥ ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ  
ΕΞΕΤΑΖΟΜΕΝΟ ΜΑΘΗΜΑ: ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ  
ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ  
ΟΜΑΔΑΣ ΠΡΟΣΑΝΑΤΟΛΙΣΜΟΥ  
ΣΠΟΥΔΩΝ ΟΙΚΟΝΟΜΙΑΣ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΣΥΝΟΛΟ ΣΕΛΙΔΩΝ: ΠΕΝΤΕ (7)  
Α Π Α Ν Τ Η Σ Ε Ι Σ

**ΘΕΜΑ Α**

- A1.** 1. Σωστό  
2. Λάθος  
3. Σωστό  
4. Σωστό  
5. Σωστό
- A2.** 1.  $K \text{ DIV } 1000$   
2.  $L \text{ MOD } 10$   
3.  $(M+=49) \text{ DIV } 50$   
4.  $(\Delta+1) \text{ MOD } 60$   
5.  $A - B * (A \text{ DIV } B)$
- A3.** α. Για να έχει έννοια κάθε σύγκριση μεταξύ δύο προγραμμάτων αλγορίθμων, θα πρέπει να ικανοποιούνται οι παρακάτω προϋποθέσεις:
- και τα δύο προγράμματα να έχουν συνταχθεί στην ίδια γλώσσα προγραμματισμού,
  - να έχει χρησιμοποιηθεί ο ίδιος μεταφραστής της γλώσσας προγραμματισμού,
  - να χρησιμοποιείται η ίδια υπολογιστική πλατφόρμα,
  - ακριβώς τα ίδια δεδομένα να αποτελούν είσοδο κατά τον έλεγχο των δύο αλγορίθμων.
- β. Τα σύγχρονα προγραμματιστικά περιβάλλοντα περιέχουν όλα τα εργαλεία που απαιτούνται για τη συγγραφή και την εκτέλεση προγραμμάτων και βοηθούν στη διόρθωση των λαθών τους. Για τη δημιουργία, τη μετάφραση και την εκτέλεση ενός προγράμματος απαιτούνται τρία προγράμματα: ο συντάκτης, ο μεταγλωττιστής και ο συνδέτης.  
Το κάθε προγραμματιστικό περιβάλλον έχει φυσικά διαφορετικά εργαλεία και ιδιότητες. Για παράδειγμα ένα περιβάλλον οπτικού (visual) προγραμματισμού πρέπει να περιέχει οπωσδήποτε και ειδικό συντάκτη που να διευκολύνει τη δημιουργία γραφικών αντικειμένων (για παράδειγμα φόρμες, λίστες, παράθυρα διαλόγου) παρέχοντας στον προγραμματιστή τα αντίστοιχα γραφικά εργαλεία.
- γ. Ένα υποπρόγραμμα που έχει γραφεί μπορεί να χρησιμοποιηθεί πολύ εύκολα και σε άλλα προγράμματα. Από τη στιγμή που έχει δημιουργηθεί, η χρήση του δεν διαφέρει από τη χρήση των ενσωματωμένων συναρτήσεων που παρέχει η γλώσσα προγραμματισμού, όπως για τον υπολογισμό του ημίτονου ή του συνημίτονου ή την εντολή με την οποία εκτελεί μια συγκεκριμένη διαδικασία, για παράδειγμα γράφει στην οθόνη (εντολή γράψε). Αν, λοιπόν, χρειάζεται συχνά κάποια λειτουργία που δεν

## ΑΡΧΗ 2ΗΣ ΣΕΛΙΔΑΣ

υποστηρίζεται απευθείας από τη γλώσσα, όπως για παράδειγμα η εύρεση του μικρότερου δύο αριθμών, τότε μπορεί να γραφεί το αντίστοιχο υποπρόγραμμα. Η συγγραφή πολλών υποπρογραμμάτων και η δημιουργία βιβλιοθηκών με αυτά, ουσιαστικά επεκτείνουν την ίδια τη γλώσσα προγραμματισμού.

- A4.** Μια στοίβα μπορεί να υλοποιηθεί πολύ εύκολα με τη βοήθεια ενός μονοδιάστατου πίνακα. Μια βοηθητική μεταβλητή, *top*, χρησιμοποιείται για να δείχνει το στοιχείο που τοποθετήθηκε τελευταίο στην κορυφή της στοίβας. Για την εισαγωγή ενός νέου στοιχείου στη στοίβα (ώθηση) αρκεί να αυξηθεί η μεταβλητή *top* κατά ένα και στη θέση αυτή να εισέλθει το στοιχείο. Η διαδικασία της ώθησης πρέπει οπωσδήποτε να ελέγχει, αν η στοίβα είναι γεμάτη (υπερχείλιση)  
Το τμήμα προγράμματος για την ώθηση πολλών στοιχείων σε στοίβα είναι:

### ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ

**ΓΡΑΨΕ** "Δώστε στοιχείο για να εισαχθεί στη στοίβα"

**ΔΙΑΒΑΣΕ** στοιχείο

**ΑΝ**  $top < N$  **ΤΟΤΕ**

$top \leftarrow top + 1$

$A[top] \leftarrow$  στοιχείο

**ΑΛΛΙΩΣ**

**ΓΡΑΨΕ** 'Η στοίβα είναι γεμάτη. Υπερχείλιση.'

**ΤΕΛΟΣ\_ΑΝ**

**ΜΕΧΡΙΣ\_ΟΤΟΥ**  $top = N$

Για την εξαγωγή ενός στοιχείου από τη στοίβα (απώθηση) εξέρχεται πρώτα το στοιχείο που δείχνει η μεταβλητή *top* και στη συνέχεια η *top* μειώνεται κατά ένα για να δείχνει τη νέα κορυφή. Η διαδικασία της απώθησης ελέγχει, αν υπάρχει ένα τουλάχιστον στοιχείο στη στοίβα, δηλαδή ελέγχει αν γίνεται υποχείλιση της στοίβας.

Το τμήμα προγράμματος για την απώθηση πολλών στοιχείων από στοίβα είναι:

### ΑΡΧΗ\_ΕΠΑΝΑΛΗΨΗΣ

**ΑΝ**  $top \geq 1$  **ΤΟΤΕ**

στοιχείο  $\leftarrow A[top]$

**ΓΡΑΨΕ** στοιχείο

$top \leftarrow top - 1$

**ΑΛΛΙΩΣ**

**ΓΡΑΨΕ** 'Η στοίβα είναι άδεια. Υποχείλιση στοίβας'

**ΤΕΛΟΣ\_ΑΝ**

**ΜΕΧΡΙΣ\_ΟΤΟΥ**  $top = 0$

Ο δείκτης *top* σε μια στοίβα *N* θέσεων μπορεί να πάρει ως μέγιστη τιμή το *N* (στοίβα γεμάτη) και ως ελάχιστη τιμή το 0 (στοίβα άδεια)

- A5.**
1. 20
  2.  $\leq$
  3. M
  4. S

5. L
6. +
7. R
8. -
9. **ΑΛΗΘΗΣ**
10. M

**ΘΕΜΑ Β**

**B1. α.** Ο αλγόριθμος έχει μία επαναληπτική διαδικασία η οποία τερματίζεται με δύο τρόπους: όταν διαβαστεί ο αριθμός 0, ή όταν πραγματοποιηθούν 10 επαναλήψεις. Η χειρότερη περίπτωση για αυτόν τον αλγόριθμο προκύπτει όταν γίνουν 10 επαναλήψεις (δηλαδή μέχρι να ισχύει το  $n=0$ ). (Η χειρότερη περίπτωση αντιπροσωπεύει τις τιμές εκείνες, που, όταν δίνονται ως είσοδος στον αλγόριθμο, οδηγούν στην εκτέλεση μέγιστου αριθμού βασικών πράξεων.) Το μέγεθος του αλγορίθμου μπορεί να εκφρασθεί από τη μεταβλητή  $n$ , που στην ουσία εκφράζει το πλήθος των επαναλήψεων του βασικού βρόχου του αλγορίθμου. (Το μέγεθος ενός αλγορίθμου εκφράζεται, από κάποιες μεταβλητές, που απεικονίζουν τους διαφορετικούς συνδυασμούς τιμών που κρίνουν τη συμπεριφορά του αλγορίθμου και δίνονται ως δεδομένα στον αλγόριθμο. Τα δεδομένα συνιστούν το μέγεθος της εισόδου ενός αλγορίθμου.)

**β. Αλγόριθμος B1**

$n \leftarrow 10$

**Διάβασε**  $m$

$n \leftarrow n-1$

**Όσο** ( $m < > 0$ ) **και** ( $n < > 0$ ) **επανάλαβε**

**Διάβασε**  $m$

$n \leftarrow n-1$

**Τέλος\_επανάληψης**

**Εκτύπωσε**  $m$

**Τέλος B1**

**B2. ΣΥΝΑΡΤΗΣΗ**  $A_{\gamma 1}(\alpha, \beta)$ : **ΑΚΕΡΑΙΑ**

**ΜΕΤΑΒΛΗΤΕΣ**

**ΑΚΕΡΑΙΕΣ:**  $\alpha, \beta, \alpha\theta\rho$

**ΑΡΧΗ**

$\alpha\theta\rho \leftarrow 0$

**ΟΣΟ**  $\alpha > 0$  **ΕΠΑΝΑΛΑΒΕ**

**ΑΝ**  $\alpha \bmod 2 \neq 0$  **ΤΟΤΕ**

$\alpha\theta\rho \leftarrow \alpha\theta\rho + \beta$

**ΤΕΛΟΣ\_ΑΝ**

$A \leftarrow \alpha \text{ DIV } 2$

$\beta \leftarrow 2 * \beta$

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

$A_{\gamma 1} \leftarrow \alpha\theta\rho$

**ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ**

**ΣΥΝΑΡΤΗΣΗ**  $\text{Αλγ2}(\alpha, \beta)$ : **ΑΚΕΡΑΙΑ**  
**ΜΕΤΑΒΛΗΤΕΣ**

**ΑΚΕΡΑΙΕΣ:**  $x, y, z, \text{ΜΚΔ}$

**ΑΡΧΗ**

$z \leftarrow y$

**ΟΣΟ**  $z < > 0$  **ΕΠΑΝΑΛΑΒΕ**

$Z \leftarrow x \text{ MOD } y$

$X \leftarrow y$

$Y \leftarrow z$

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

$\text{ΜΚΔ} \leftarrow x$

$\text{Αλγ2} \leftarrow \text{ΜΚΔ}$

**ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ**

**ΘΕΜΑ Δ**

**ΠΡΟΓΡΑΜΜΑ** Λειτουργικά\_Συστήματα

**ΜΕΤΑΒΛΗΤΕΣ**

**ΠΡΑΓΜΑΤΙΚΕΣ:**  $\text{ΒΑΘ}[5]$ , όχι\_τελική

**ΑΚΕΡΑΙΕΣ:**  $i$ , Συμτεξ, Συμοχι

**ΧΑΡΑΚΤΗΡΕΣ:** Ον

**ΑΡΧΗ**

$\text{Συμτεξ} \leftarrow 0$

$\text{Συμοχι} \leftarrow 0$

**ΓΡΑΨΕ** 'Δώστε ονοματεπώνυμο φοιτητή'

**ΔΙΑΒΑΣΕ** Ον

**ΟΣΟ** Ον  $< > '*'$  **ΕΠΑΝΑΛΑΒΕ**

**ΓΙΑ**  $i$  **ΑΠΟ** 1 **ΜΕΧΡΙ** 4

**ΓΡΑΨΕ** 'Δώστε βαθμούς ασκήσεων, τρεις (3) συνολικά, και προόδου'

**ΚΑΛΕΣΕ** Διαβαθμ( $\text{ΒΑΘ}[i]$ )

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΑΝ**  $(\text{ΒΑΘ}[1] + \text{ΒΑΘ}[2] + \text{ΒΑΘ}[3] + \text{ΒΑΘ}[4]) / 4 \geq 5$  **ΤΟΤΕ**

**ΓΡΑΨΕ** 'Δώστε βαθμό τελικής εξέτασης'

**ΚΑΛΕΣΕ** Διαβαθμ( $\text{ΒΑΘ}[5]$ )

**ΓΡΑΨΕ** ΤελικόςΒαθμός ( $\text{ΒΑΘ}[1], \text{ΒΑΘ}[2], \text{ΒΑΘ}[3], \text{ΒΑΘ}[4], \text{ΒΑΘ}[5]$ )

$\text{Συμτεξ} \leftarrow \text{Συμτεξ} + 1$

**ΑΛΛΙΩΣ**

**ΓΡΑΨΕ** 'Ο/Η', Ον, 'δεν δικαιούται να συμμετέχει στην τελική εξέταση'

$\text{Συμοχι} \leftarrow \text{Συμοχι} + 1$

**ΤΕΛΟΣ\_ΑΝ**

**ΓΡΑΨΕ** 'Δώστε ονοματεπώνυμο επόμενου φοιτητή'

**ΔΙΑΒΑΣΕ** Ον

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

όχι\_τελική  $\leftarrow 100 * \text{Συμοχι} / (\text{Συμτεξ} + \text{Συμοχι})$

**ΓΡΑΨΕ** 'Ποσοστό μη δικαιούχων στην τελική εξέταση', όχι\_τελική, '%'

**ΤΕΛΟΣ\_ΠΡΟΓΡΑΜΜΑΤΟΣ**

.....  
**ΔΙΑΔΙΚΑΣΙΑ** Διαβαθμ(β)

**ΜΕΤΑΒΛΗΤΕΣ**

**ΠΡΑΓΜΑΤΙΚΕΣ:** β

**ΑΡΧΗ**

**ΔΙΑΒΑΣΕ** β

**ΟΣΟ** β < 0 **Ή** β > 10 **ΕΠΑΝΑΛΑΒΕ**

**ΓΡΑΨΕ** 'Πρέπει β >= 0 και β <= 10'

**ΔΙΑΒΑΣΕ** β

**ΤΕΛΟΣ\_ΕΠΑΝΑΛΗΨΗΣ**

**ΤΕΛΟΣ\_ΔΙΑΔΙΚΑΣΙΑΣ**

.....  
**ΣΥΝΑΡΤΗΣΗ** ΤελικοςΒαθμός(α1, α2, α3, π, τεξ): **ΠΡΑΓΜΑΤΙΚΗ**

**ΜΕΤΑΒΛΗΤΕΣ**

**ΠΡΑΓΜΑΤΙΚΕΣ:** α1, α2, α3, π, τεξ

**ΑΡΧΗ**

**ΑΝ** τεξ < 5 **ΤΟΤΕ**

ΤελικοςΒαθμός  $\leftarrow 0.1 * (\alpha_1 + \alpha_2 + \alpha_3) + 0.15 * \pi + 0.55 * \text{τεξ}$

**ΑΛΛΙΩΣ**

ΤελικοςΒαθμός  $\leftarrow 0.06 * (\alpha_1 + \alpha_2 + \alpha_3) + 0.12 * \pi + 0.7 * \text{τεξ}$

**ΤΕΛΟΣ\_ΑΝ**

**ΤΕΛΟΣ\_ΣΥΝΑΡΤΗΣΗΣ**

**ΘΕΜΑ Δ**

**Αλγόριθμος** Μπάσκειτ

**Για** i από 1 μέχρι 20

**Διάβασε** O[i]

**Τέλος\_επανάληψης**

**Για** i από 1 μέχρι 20

**Για** j από 1 μέχρι 40

**Διάβασε** Π[i, j]

**Τέλος\_επανάληψης**

**Τέλος\_επανάληψης**

πόσοι  $\leftarrow 0$

**Για** j από 1 μέχρι 40

max  $\leftarrow \Pi[1, j]$

**Για** i από 1 μέχρι 20

**Αν** Π[i, j] > max **τότε**

```

    max ← Π[i, j]
    Τέλος_αν
Τέλος_επανάληψης
α ← 0
Για i από 1 μέχρι 20
    Αν Π[i, j] = max τότε
        α ← α + 1
    Τέλος_αν
Τέλος_επανάληψης
Αν α = 1 τότε ! μόνο 1 max
    πόσοι ← πόσοι + 1
Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε πόσοι
Για i από 1 μέχρι 20
    done ← ψευδής
    j ← 1
    Όσο done = ψευδής και j ≤ 40 επανάλαβε
        Αν Π[i, j] = -1 τότε
            done ← αληθής
        αλλιώς
            j ← j + 1
        Τέλος_αν
    Τέλος_επανάληψης
    Αν done = ψευδής τότε
        Εμφάνισε O[i]
    Τέλος_αν
Τέλος_επανάληψης
Διάβασε όνομα ! ερώτημα Γ5
done ← ψευδής
pos ← 0
i ← 1
Όσο done = ψευδής και i ≤ 20 επανάλαβε
    Αν O[i] = όνομα τότε
        done ← αληθής
        pos ← i
    αλλιώς
        i ← i + 1
    Τέλος_αν
Τέλος_επανάληψης
Αν done = αληθής τότε
    Για i από 1 μέχρι 40

```

ΑΓ[i] ← i  
Τέλος\_επανάληψης  
Για i από 2 μέχρι 40  
  Για j από 40 μέχρι i με\_βήμα -1  
    Αν Π[ros, j-1] < Π[ros, j] τότε  
      Αντιμετάθεσε Π[ros, j-1], Π[ros, j]  
      Αντιμετάθεσε ΑΓ[j-1], ΑΓ[j]  
    Τέλος\_αν  
  Τέλος\_επανάληψης  
Τέλος\_επανάληψης  
Αν Π[ros, 6] < > -1 τότε ! έχει 6 καλύτερους  
  Εμφάνισε "Οι 6 καλύτεροι αγώνες για τον παίκτη", όνομα  
  Για j από 1 μέχρι 6  
    Εμφάνισε ΑΓ[j]  
  Τέλος\_επανάληψης  
αλλιώς  
  Εμφάνισε " δεν έχει καν συμμετοχή σε 6 αγώνες"  
Τέλος\_αν  
αλλιώς  
  Εμφάνισε "Δεν υπάρχει παίκτης με αυτό το όνομα"  
Τέλος\_αν  
Τέλος Μπάσκειτ

ΕΠΙΜΕΛΕΙΑ ΘΕΜΑΤΩΝ : ΚΑΛΟΚΑΙΡΙΝΟΣ ΛΑΜΠΗΣ- ΑΕΠΠ

ΦΡΟΝΤΙΣΤΗΡΙΑ ΟΡΟΣΗΜΟ

ΑΓΙΑ ΠΑΡΑΣΚΕΥΗ – ΠΑΠΑΓΟΥ-ΧΟΛΑΡΓΟΣ