ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

ΕΠΑΝΑΛΗΠΤΙΚΟ ΔΙΑΓΩΝΙΣΜΑ

ΣΧΟΛΙΚΟΥ ΕΤΟΥΣ 2014-2015

Επιμέλεια:

Ομάδα Διαγωνισμάτων από το "Στέκι των Πληροφορικών"

ΕΝΔΕΙΚΤΙΚΕΣ ΛΥΣΕΙΣ

Θέμα Α

Α1. α.

1. ΣΩΣΤΟ

2. ΣΩΣΤΟ

3. ΛΑΘΟΣ

4. ΛΑΘΟΣ

β. Ακολουθεί αιτιολόγηση για κάθε μια από τις προτάσεις:

1. Σε αντίθεση με τα ανοιχτά προβλήματα για τα οποία δεν έχει αποδειχθεί ότι δεν επιδέχονται λύση, στα άλυτα προβλήματα έχουμε πλέον φτάσει σε αυτήν την παραδοχή. Επομένως, δεν πρόκειται ποτέ να βρεθεί κάποια λύση και έτσι να χαρακτηριστούν επιλύσιμα.

2. Κάθε φορά που καλείται κάποιο υποπρόγραμμα τότε ωθείται στη στοίβα χρόνου εκτέλεσης η αντίστοιχη διεύθυνση επιστροφής. Μετά την εκτέλεση του υποπρογράμματος η διεύθυνση απωθείται από τη στοίβα. Αν η στοίβα χρόνου εκτέλεσης περιέχει δύο τιμές, αυτό σημαίνει ότι πρέπει να γίνουν δύο απωθήσεις για να μεταφερθεί ο έλεγχος στο κύριο πρόγραμμα. Επομένως, τη συγκεκριμένη χρονική στιγμή ο έλεγχος βρίσκεται σε ένα υποπρόγραμμα που έχει κληθεί από ένα άλλο υποπρόγραμμα το οποίο έχει κληθεί από το κύριο πρόγραμμα.

3. Αν ο αρχικός αριθμός είναι το 17 τότε κάνοντας ολίσθηση προς τα δεξιά, δηλαδή ακέραια διαίρεση δια δύο, προκύπτει ο αριθμός 8. Αν σε αυτόν εφαρμόσουμε ολίσθηση προς τα αριστερά, δηλαδή πολλαπλασιασμό επί δύο, τότε ο τελικός αριθμός θα είναι το 16. Κάτι παρόμοιο θα συμβαίνει για κάθε περιττό αριθμό. Άρα ο τελικός αριθμός δεν θα είναι πάντα ίσος με τον αρχικό.

4. Αφού πρώτα πρέπει να ακυρώνεται η τελευταία ενέργεια που έχει πραγματοποιηθεί, μετά η προτελευταία, κ.ο.κ., τότε η κατάλληλη δομή δεδομένων είναι η στοίβα. Έτσι κάθε ενέργεια θα προκαλεί ώθηση και κάθε αναίρεση απώθηση από τη στοίβα. Αν είχε χρησιμοποιηθεί η ουρά τότε πρώτα θα ακυρωνόταν η πρώτη ενέργεια που θα είχε πραγματοποιηθεί, μετά η δεύτερη, κ.ο.κ.

Α2.

1. +
2. Β
3. Α
4. Β
5. Α
6. -

Α3.

1. `Ν - 1`
2. `1`
3. `-1`
4. `i`
5. `1`
6. `j`
7. `j`
8. `j + 1`

Παρατήρηση: Οι παραπάνω απαντήσεις 7 και 8 μπορούν να δοθούν και με την αντίθετη σειρά.

Α4.

α. Το πρώτο συντακτικό λάθος είναι ότι η μεταβλητή `θ` δεν έχει δηλωθεί. Θα πρέπει να δηλωθεί ως ακέραια αφού εκχωρούνται σε αυτήν ακέραιες τιμές.

Το δεύτερο συντακτικό λάθος είναι η χρήση της λέξης `ΤΟΤΕ` αντί της `ΕΠΑΝΑΛΑΒΕ` στην εντολή ΟΣΟ.

β. Το πρώτο λογικό λάθος είναι ότι η αύξηση κατά ένα της μεταβλητής `π` (που εκχωρείται στη μεταβλητή `θ` για τον προσδιορισμό της σειράς με την οποία διαβάστηκε ο μικρότερος βαθμός) δεν γίνεται κάθε φορά που διαβάζεται νέος αριθμός. Θα πρέπει η εντολή `π<-π+1` να μετακινηθεί μετά το τέλος της εντολής ΑΝ.

Το δεύτερο λογικό λάθος είναι ότι στην περίπτωση που όλοι οι βαθμοί που δοθούν είναι ίσοι με τη μέγιστη δυνατή τιμή, δηλαδή ίσοι με 100, τότε η συνθήκη της εντολής ΑΝ δεν θα γίνει ποτέ αληθής οπότε η μεταβλητή `θ` θα παραμείνει απροσδιόριστη. Μια λύση είναι η αρχική τιμή που εκχωρείται στη μεταβλητή `min` να είναι μεγαλύτερη από 100, π.χ. το 101, οπότε σίγουρα στην πρώτη επανάληψη η συνθήκη της εντολής ΑΝ θα είναι αληθής. Άλλη λύση είναι πριν την εντολή ΟΣΟ να εκχωρείται στη μεταβλητή `θ` το 1.

γ. Τα συντακτικά λάθη εντοπίζονται και διορθώνονται ευκολότερα. Ο λόγος είναι ότι αυτά ανιχνεύονται από το μεταγλωττιστή (ή το διερμηνευτή) που επιπρόσθετα εμφανίζει κατάλληλα διαγνωστικά μηνύματα. Αντίθετα τα λογικά λάθη εμφανίζονται κατά την εκτέλεση του προγράμματος και, όπως φάνηκε στο παραπάνω παράδειγμα, μπορεί να αφορούν σπάνιες περιπτώσεις δεδομένων εισόδου, οπότε η ύπαρξή τους να μη γίνει έγκαιρα αντιληπτή ή να μη γίνει ποτέ.

Α5.

α. Εμφανίζονται 4 χαρακτήρες "*".

β. Εμφανίζονται 10 χαρακτήρες "+". *[εξήγηση: 1+2+3+4]*

γ. Εμφανίζονται 1024 χαρακτήρες "#". *[εξήγηση: 4+10+10*101]*

Θέμα Β

Β1.

αριθμός γραμμής	**συνθήκη 1: i≤N και j≤M**	**συνθήκη 2: A[i]<B[j]**	**i**	**j**	**K**	**Γ[1]**	**Γ[2]**	**Γ[3]**	**Γ[4]**
1			1						
2				1					
3					0				
4	ΑΛΗΘΗΣ								
5					1				
6		ΨΕΥΔΗΣ							
10						1			
11				2					
4	ΑΛΗΘΗΣ								
5					2				
6		ΑΛΗΘΗΣ							
7							3		
8			2						
4	ΑΛΗΘΗΣ								
5					3				
6		ΨΕΥΔΗΣ							
10								4	
11				3					
4	ΑΛΗΘΗΣ								
5					4				
6		ΨΕΥΔΗΣ							
10									6
11				4					
4	ΨΕΥΔΗΣ								

Β2.

1. `k + 1`

2. `N + M`

3. `i > N` (εναλλακτικά: `j ≤ M`)

4. `Γ`

B3.

```
i ← 1
j ← 1
k ← 0
Αρχή_επανάληψης
  k ← k + 1
  Αν Α[i] < B[j] τότε
    Γ[k] ← A[i]
    i ← i + 1
  αλλιώς
    Γ[k] ← B[j]
    j ← j + 1
  Τέλος_αν
Μέχρις_ότου i > N ή j > M
```

Παρατήρηση: Θεωρούμε ότι και οι δύο πίνακες Α και Β θα έχουν αρχικά τουλάχιστον ένα στοιχείο ο καθένας!

B4.

α. Στην εντολή 6 κάθε φορά συγκρίνονται μεταξύ τους τα A[i] και B[j] και στη συνέχεια το μικρότερο από αυτά αντιγράφεται στο Γ[k]. Επομένως οι αρχικοί πίνακες Α και Β θα πρέπει αρχικά να είναι ταξινομημένοι με αύξουσα διάταξη.

β. Αν οι αρχικοί πίνακες Α και Β είναι ταξινομημένοι με φθίνουσα διάταξη και θεωρώντας ότι και ο τελικός πίνακας Γ θα πρέπει να είναι ταξινομημένος ομοίως, τότε αρκεί να αλλάξει η συνθήκη της εντολής 6 από `A[i]<B[j]` σε `A[i]>B[j]`.

Θέμα Γ

```
ΑΛΓΟΡΙΘΜΟΣ Εκλογή
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 18
   ΔΙΑΒΑΣΕ ΥΠΟΨ[i]
   ΠΡΩΤΙΕΣ[i] ← 0
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

πλ_ψηφ ← 0
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 18
    ΨΗΦΟΙ[i] ← 0
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

  ΓΙΑ k ΑΠΟ 1 ΜΕΧΡΙ 115
    ΔΙΑΒΑΣΕ ΟΝΟΜΑ
    βρέθηκε ← ψευδής
    i ← 1
    ΟΣΟ βρέθηκε = ψευδής ΕΠΑΝΑΛΑΒΕ
      ΑΝ ΥΠΟΨ[i] = ΟΝΟΜΑ ΤΟΤΕ
```

```
        βρέθηκε ← αληθής
        θ ← i
      ΑΛΛΙΩΣ
        i ← i+1
      ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

    ΨΗΦΟΙ[θ] ← ΨΗΦΟΙ[θ] + 1
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

  max ← ΨΗΦΟΙ[1]
  θmax ← 1
  ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 18
    ΑΝ ΨΗΦΟΙ[i] > max ΤΟΤΕ
      max ← ΨΗΦΟΙ[i]
      θmax ← i
    ΤΕΛΟΣ_ΑΝ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

  ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 18
    ΑΝ ΨΗΦΟΙ[i] = max ΤΟΤΕ ΠΡΩΤΙΕΣ[i] ← ΠΡΩΤΙΕΣ[i] + 1
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

  ΑΝ max < 115*2/3 ΤΟΤΕ
    ΕΜΦΑΝΙΣΕ "Μαύρος καπνός"
    done ← ψευδής
  ΑΛΛΙΩΣ
    ΕΜΦΑΝΙΣΕ "Λευκός καπνός"
    done ← αληθής
  ΤΕΛΟΣ_ΑΝ

  πλ_ψηφ ← πλ_ψηφ + 1
ΜΕΧΡΙΣ_ΟΤΟΥ done

ΕΜΦΑΝΙΣΕ ΥΠΟΨ[θmax]

ΑΝ ΠΡΩΤΙΕΣ[θmax] = πλ_ψηφ ΤΟΤΕ
  ΕΜΦΑΝΙΣΕ "Ο Πάπας ήταν το φαβορί"
ΑΛΛΙΩΣ
  ΕΜΦΑΝΙΣΕ "Αουτσάιντερ"
ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ Εκλογή
```

Θέμα Δ

```
ΠΡΟΓΡΑΜΜΑ Μετάδοση_θερμότητας
ΜΕΤΑΒΛΗΤΕΣ
  ΑΚΕΡΑΙΕΣ: i, j, k, Δευτ
  ΠΡΑΓΜΑΤΙΚΕΣ: Θ[1000, 1000], ΠΘ[1000, 1000], θπ, θτ, Αθρ, ΜΟ
  ΛΟΓΙΚΕΣ: Ισχύει
ΑΡΧΗ

ΔΙΑΒΑΣΕ θπ
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 1000
  ΓΙΑ j ΑΠΌ 1 ΜΕΧΡΙ 1000
    Θ[i, j] <- θπ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΔΙΑΒΑΣΕ θτ
ΜΕΧΡΙΣ_ΟΤΟΥ θτ >= θπ + 50
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 1000
  Θ[i, 1] <- θτ
  Θ[i, 1000] <- θτ
  Θ[1, i] <- θτ
  Θ[1000, i] <- θτ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Δευτ <- 0
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 999
    ΓΙΑ j ΑΠΟ 2 ΜΕΧΡΙ 999
       ΠΘ[i, j] <- Νέα_θερμοκρασία(Θ, i, j)
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

  Αθρ <- 0
  ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 999
    ΓΙΑ j ΑΠΟ 2 ΜΕΧΡΙ 999
       Θ[i, j] <- ΠΘ[i, j]
       Αθρ <- Αθρ + Θ[i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΜΟ <- Αθρ / (998 * 998)
  ΓΡΑΨΕ ΜΟ
  Δευτ <- Δευτ + 1
ΜΕΧΡΙΣ_ΟΤΟΥ θτ - ΜΟ < 0.5

ΓΡΑΨΕ Δευτ
```

```
Ισχύει <- ΑΛΗΘΗΣ
i <- 2
k <- 502
ΟΣΟ i <= 500 ΚΑΙ Ισχύει = ΑΛΗΘΗΣ ΕΠΑΝΑΛΑΒΕ
  ΑΝ Θ[i-1, i-1] < Θ[i, i] Ή Θ[k-1, k-1] > Θ[k, k] ΤΟΤΕ
    Ισχύει <- ΨΕΥΔΗΣ
  ΑΛΛΙΩΣ
    i <- i + 1
    k <- k + 1
  ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ Ισχύει = ΑΛΗΘΗΣ ΤΟΤΕ
  ΓΡΑΨΕ 'Η θερμότητα έχει μεταδοθεί ομοιόμορφα στην πλάκα'
ΑΛΛΙΩΣ
  ΓΡΑΨΕ 'Η θερμότητα δεν έχει μεταδοθεί ομοιόμορφα στην πλάκα'
ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ


ΣΥΝΑΡΤΗΣΗ Νέα_θερμοκρασία(Θ, γρ, στ): ΠΡΑΓΜΑΤΙΚΗ
ΜΕΤΑΒΛΗΤΕΣ
  ΠΡΑΓΜΑΤΙΚΕΣ: Θ[1000, 1000], Αθρ
  ΑΚΕΡΑΙΕΣ: γρ, στ, i, j
ΑΡΧΗ
  Αθρ <- 0
  ΓΙΑ i ΑΠΟ γρ - 1 ΜΕΧΡΙ γρ + 1
    ΓΙΑ j ΑΠΟ στ - 1 ΜΕΧΡΙ στ + 1
      Αθρ <- Αθρ + Θ[i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  Αθρ <- Αθρ + Θ[γρ, στ]
  Νέα_θερμοκρασία <- Αθρ / 10
ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ
```