

# **Ε.Α.Π./ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ**

## **3η ΓΡΑΠΤΗ ΕΡΓΑΣΙΑ**

**ΑΚΑΔΗΜΑΪΚΟΥ ΕΤΟΥΣ 2014-2015**

**4<sup>ος</sup> Τόμος**

## **ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ**

**24/1/2015**

**Ημερομηνία παράδοσης εργασίας: Παρασκευή 27/2/2015**

**Καταληκτική ημερομηνία παραλαβής: Τετάρτη 4/3/2015<sup>1</sup>**

**Ημερομηνία ανάρτησης ενδεικτικών λύσεων: Σάββατο 7/3/2015**

**Καταληκτική ημερομηνία αποστολής σχολίων στο φοιτητή: Κυριακή 22/3/2015**

**ΥΠΟΕΡΓΑΣΙΑ 1.** (βαθμοί 25)

Εισαγωγικά θέματα, αναδρομή

**ΥΠΟΕΡΓΑΣΙΑ 2.** (βαθμοί 30)

Αμυντικός προγραμματισμός, πίνακες

**ΥΠΟΕΡΓΑΣΙΑ 3.** (βαθμοί 25)

Χειρισμός πινάκων με χρήση δεικτών

**ΥΠΟΕΡΓΑΣΙΑ 4.** (βαθμοί 20)

Δισδιάστατοι πίνακες

**ΣΥΝΟΛΟ** (βαθμοί 100)

---

<sup>1</sup> Σύμφωνα με τον Κανονισμό Σπουδών, η καταληκτική ημερομηνία για την παραλαβή της Γ.Ε. από το μέλος ΣΕΠ είναι η επόμενη Τετάρτη από το τέλος της εβδομάδας παράδοσης Γ.Ε.

**ΥΠΟΕΡΓΑΣΙΑ 1.**

(βαθμοί 25)

**A.** Αν  $a=6$ , τι θα τυπωθεί στην οθόνη μετά την ακολουθιακή εκτέλεση των εξής εντολών:


```
printf("%d\n", ++a);  
printf("%d\n", a--);  
printf("%d\n", a<<2);  
printf("%d\n", a>>1);  
printf("%d\n", a^a);
```

Τεκμηριώστε την απάντησή σας περιγράφοντας εντολή προς εντολή την εκτέλεση του κώδικα.

**B.** Το ακόλουθο πρόγραμμα έχει σκοπό τη μετατροπή ενός αριθμού ημερών σε χρόνια, μήνες και ημέρες. Συμπληρώστε το πρόγραμμα στο σημείο που συμβολίζεται με «???» ώστε να λειτουργεί σωστά. Υποθέστε ότι ο χρόνος έχει 360 ημέρες, ενώ κάθε μήνας έχει 30 ημέρες.

```
int main()  
{  
    int n;  
    printf("Give the number of days\n");  
    scanf("%d", &n);  
    printf("%d year(s), %d month(s), %d day(s)", ???);  
    return (0);  
}
```

Ακολουθεί ένα παράδειγμα εκτέλεσης (Εικόνα 1).



```
Give the number of days  
543  
1 year(s), 6 month(s), 3 day(s)
```

**Εικόνα 1**

**Γ.** Έστω η εξής συνάρτηση στη γλώσσα προγραμματισμού C:

```
void func(int k)  
{  
    if (k==0)  
        return;  
    printf("%d", k-1);  
    int i;  
    for (i=1; i<k; i++)  
        func(i);  
}
```

Τι θα τυπωθεί στην οθόνη μετά την ακόλουθη κλήση της συνάρτησης:

```
func(4);
```

Τεκμηριώστε την απάντησή σας περιγράφοντας εντολή προς εντολή την εκτέλεση της συνάρτησης.

**Δ.** Ο υπολογισμός του ετήσιου φόρου 20 μισθωτών υπαλλήλων γίνεται με βάση το ετήσιο ακαθάριστο εισόδημά τους ως εξής:

- Φόρος 22% για εισόδημα μέχρι και 25000 €

- Φόρος 32% για τα επόμενα 17000 €
- Φόρος 42% για το υπόλοιπο εισόδημα

Να γράψετε πρόγραμμα σε γλώσσα C, το οποίο:

- Διαβάζει τις ετήσιες ακαθάριστες αποδοχές όλων των υπαλλήλων με χρήση αμυντικού προγραμματισμού (δηλ. δεν πρέπει να γίνονται δεκτές αρνητικές τιμές).
- Εκτυπώνει τον ετήσιο φόρο για κάθε υπάλληλο.
- Εκτυπώνει το μηνιαίο καθαρό μισθό κάθε υπαλλήλου (θεωρώντας 12 μισθούς ανά έτος).
- Εκτυπώνει το μεγαλύτερο καθαρό ετήσιο εισόδημα και τον αντίστοιχο φόρο.

## ΥΠΟΕΡΓΑΣΙΑ 2.

(βαθμοί 30)

Μεγάλη πολυεθνική φαρμακοβιομηχανία επιθυμεί να επεκταθεί ανοίγοντας ένα νέο εργοστάσιο κατασκευής φαρμάκων στην Ευρώπη. Η επιλογή της χώρας εγκατάστασης θα εξαρτηθεί από τη συνολική σταθμισμένη βαθμολογία της σε σχέση με ένα σύνολο κριτηρίων που ενδιαφέρουν την επιχείρηση. Στον Πίνακα 1 φαίνονται τα κριτήρια, οι συντελεστές βαρύτητας και η βαθμολογία ανά κριτήριο για τρεις υποψήφιες χώρες. Ο τελικός βαθμός για κάθε χώρα που εμφανίζεται επίσης στον Πίνακα 1 προκύπτει από τη σχέση της εξίσωσης (1):

$$\text{Τελικός Βαθμός} = \sum_{i=1}^8 w_i f_i = w_1 f_1 + w_2 f_2 + \dots + w_8 f_8 \quad (1)$$

Το  $w_i$  στην Εξίσωση (1) αντιστοιχεί στο συντελεστή βαρύτητας του κριτηρίου  $i$  (για  $i=1,\dots,8$ ) και το  $f_i$  στο βαθμό που έλαβε η χώρα σε σχέση με το κριτήριο  $i$ . Η χώρα που τελικά θα επιλεγεί είναι αυτή με το μεγαλύτερο τελικό βαθμό.

| Κριτήριο                    | Συντελεστής Βαρύτητας | Περιγραφή κριτηρίου                  | Χώρα 1      | Χώρα 2      | Χώρα 3      |
|-----------------------------|-----------------------|--------------------------------------|-------------|-------------|-------------|
| 1                           | 10%                   | Κυβερνητική σταθερότητα              | 4           | 5           | 5           |
| 2                           | 20%                   | Σταθερότητα εθνικού νομίσματος       | 5           | 5           | 5           |
| 3                           | 15%                   | Φορολογική πολιτική                  | 5           | 4           | 4           |
| 4                           | 20%                   | Εγγύτητα σε αγορές                   | 4           | 5           | 5           |
| 5                           | 5%                    | Εγγύτητα σε προμηθευτές              | 4           | 3           | 3           |
| 6                           | 5%                    | Μορφωτικό επίπεδο εργαζομένων        | 4           | 4           | 3           |
| 7                           | 15%                   | Κουλτούρα, νοοτροπία, παραγωγικότητα | 4           | 3           | 4           |
| 8                           | 10%                   | Κόστος εργασίας                      | 4           | 4           | 5           |
| <b>Άθροισμα συντελεστών</b> | <b>100%</b>           | <b>Τελικός Βαθμός</b>                | <b>4.35</b> | <b>4.30</b> | <b>4.50</b> |

**Πίνακας 1**

Ζητείται να υλοποιηθεί πρόγραμμα στη γλώσσα προγραμματισμού C με τα εξής χαρακτηριστικά:

**A.** (i) Να ορίζει το πλήθος  $N$  των κριτηρίων και τον αριθμό  $M$  των χωρών για εγκατάσταση ως *συμβολικών σταθερών* στο πρόγραμμα. (ii) Να ορίζει ένα μονοδιάστατο πίνακα αλφαριθμητικών με όνομα *criterion* ο οποίος θα περιέχει τις περιγραφές των κριτηρίων του Πίνακα 1. (iii) Να ορίζει ένα μονοδιάστατο πίνακα με όνομα *weights* που θα περιέχει τους συντελεστές βαρύτητας κάθε κριτηρίου. Κάθε θέση του *weights* θα αντιστοιχεί σε ένα κριτήριο. (iv) Να ορίζει έναν πίνακα με όνομα *scores* διαστάσεων  $N \times M$  που θα περιέχει τη βαθμολογία κάθε χώρας σε σχέση με κάθε κριτήριο. (v) Να ορίζει ένα μονοδιάστατο πίνακα με όνομα *finalScores* που θα διατηρεί τον τελικό βαθμό για κάθε χώρα, όπως υπολογίζεται από την εξίσωση (1). Τονίζεται ότι οι πίνακες *finalScores*, *weights* και *scores* απαιτείται να οριστούν ως καθολικές μεταβλητές στο πρόγραμμα.

**B.** Να γραφεί η συνάρτηση:

```
void read_weights()
```

που θα διαβάζει από το πληκτρολόγιο τους συντελεστές βαρύτητας του πίνακα *weights*. Κάθε συντελεστής βαρύτητας θα είναι ένας μη αρνητικός ακέραιος αριθμός μικρότερος του 100. Επίσης, το άθροισμα των συντελεστών πρέπει να είναι ίσο με 100. Εφαρμόστε αμυντικό προγραμματισμό στην εισαγωγή των συντελεστών.

**Γ.** Να γραφεί η συνάρτηση:

```
void read_scores()
```

που θα διαβάζει από το πληκτρολόγιο τις βαθμολογίες κάθε χώρας για κάθε κριτήριο και θα τις καταχωρεί στον πίνακα *scores*. Η ανάγνωση και καταχώρηση των βαθμών στον πίνακα να γίνεται κατά στήλη. Δηλαδή, πρώτα να διαβαστούν οι βαθμοί κάθε κριτηρίου για την 1<sup>η</sup> χώρα, μετά οι βαθμοί για τη 2<sup>η</sup> χώρα, κ.ο.κ. Κάθε βαθμός θα είναι ακέραιος θετικός αριθμός μικρότερος ή ίσος του 5. Εφαρμόστε αμυντικό προγραμματισμό στην εισαγωγή των βαθμών.

**Δ.** Να γραφεί η συνάρτηση:

```
void best_country()
```

που θα υπολογίζει τον τελικό βαθμό για κάθε χώρα σύμφωνα με την εξίσωση (1) και θα τον αποθηκεύει στον πίνακα *finalScores*. Στο τέλος η συνάρτηση να τυπώνει στην οθόνη τον τελικό βαθμό κάθε χώρας καθώς και τη χώρα με το μεγαλύτερο τελικό βαθμό. Σε περίπτωση ισοβαθμίας να τυπώνονται όλες οι χώρες που ισοβαθμούν.

Η συνάρτηση *main()* απλώς θα καλεί, κατάλληλα, όλες αυτές τις συναρτήσεις με τη σειρά που περιγράφηκαν. Ακολουθεί ένα παράδειγμα εκτέλεσης για  $N=8$  και  $M=2$  (Εικόνα 2).

|                                                                                   |  |
|-----------------------------------------------------------------------------------|--|
| Συντελεστής βαρύτητας για το κριτήριο 1 (Κυβερνητική σταθερότητα):10              |  |
| SUM(wj)= 10                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 2 (Σταθερότητα εθνικού νομίσματος):11       |  |
| SUM(wj)= 21                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 3 (Φορολογική πολιτική):12                  |  |
| SUM(wj)= 33                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 4 (Εγγύτητα σε αγορές):13                   |  |
| SUM(wj)= 46                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 5 (Εγγύτητα σε προμηθευτές):14              |  |
| SUM(wj)= 60                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 6 (Μορφωτικό επίπεδο εργαζομένων):15        |  |
| SUM(wj)= 75                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 7 (Κουλτούρα, νοοτροπία, παραγωγικότητα):16 |  |
| SUM(wj)= 91                                                                       |  |
| Συντελεστής βαρύτητας για το κριτήριο 8 (Κόστος εργασίας):9                       |  |
| SUM(wj)= 100                                                                      |  |
| -----                                                                             |  |
| Βαθμός στο κριτήριο 1 για τη χώρα 1: 1                                            |  |
| Βαθμός στο κριτήριο 2 για τη χώρα 1: 2                                            |  |
| Βαθμός στο κριτήριο 3 για τη χώρα 1: 3                                            |  |
| Βαθμός στο κριτήριο 4 για τη χώρα 1: 4                                            |  |
| Βαθμός στο κριτήριο 5 για τη χώρα 1: 5                                            |  |
| Βαθμός στο κριτήριο 6 για τη χώρα 1: 5                                            |  |
| Βαθμός στο κριτήριο 7 για τη χώρα 1: 4                                            |  |
| Βαθμός στο κριτήριο 8 για τη χώρα 1: 3                                            |  |
| -----                                                                             |  |
| Βαθμός στο κριτήριο 1 για τη χώρα 2: 2                                            |  |
| Βαθμός στο κριτήριο 2 για τη χώρα 2: 3                                            |  |
| Βαθμός στο κριτήριο 3 για τη χώρα 2: 4                                            |  |
| Βαθμός στο κριτήριο 4 για τη χώρα 2: 5                                            |  |
| Βαθμός στο κριτήριο 5 για τη χώρα 2: 5                                            |  |
| Βαθμός στο κριτήριο 6 για τη χώρα 2: 4                                            |  |
| Βαθμός στο κριτήριο 7 για τη χώρα 2: 3                                            |  |
| Βαθμός στο κριτήριο 8 για τη χώρα 2: 2                                            |  |
| -----                                                                             |  |
| Τελικός βαθμός για τη χώρα 1: 3.56                                                |  |
| Τελικός βαθμός για τη χώρα 2: 3.62                                                |  |
| Η χώρα με την υψηλότερη βαθμολογία είναι η: 2                                     |  |

Εικόνα 2

**ΥΠΟΕΡΓΑΣΙΑ 3.**

(βαθμοί 25)

Η γενική μορφή ενός πολυωνύμου βαθμού  $n$  μιας μεταβλητής  $x$  μπορεί να γραφεί ως:

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + a_{n-2} x^{n-2} + \dots + a_2 x^2 + a_1 x + a_0 \quad (2)$$

όπου  $a_i$  (για  $i=0..n$ ), είναι οι σταθεροί συντελεστές του πολυωνύμου. Σε πρώτη ανάγνωση, για τον υπολογισμό της τιμής του  $p(x)$ , για δοσμένη τιμή της μεταβλητής  $x$ , απαιτείται ο υπολογισμός των δυνάμεων της  $x$  έως το βαθμό  $n$ . Το Σχήμα Horner μας επιτρέπει να αποφύγουμε τους υπολογισμούς αυτούς καθώς η εξίσωση (2) μπορεί να γραφεί ως:

$$p(x) = (((a_n x + a_{n-1})x + a_{n-2})x + \dots + a_2)x + a_1)x + a_0 \quad (3)$$

Έτσι, κάθε φορά έχουμε έναν απλό υπολογισμό που βασίζεται στο εξής σχήμα: πολλαπλασίασε το συντελεστή του μεγιστοβάθμιου όρου (δηλαδή το  $a_n$ ) με το  $x$  και πρόσθεσε το συντελεστή της αμέσως μικρότερης τάξης (δηλαδή το  $a_{n-1}$ ). Το εξαγόμενο των πράξεων αυτών πολλαπλασίασε το με το  $x$  και πρόσθεσε το συντελεστή της αμέσως μικρότερης τάξης (δηλαδή το  $a_{n-2}$ ). Συνέχισε με τον ίδιο τρόπο, έως ότου προστεθεί ο συντελεστής μηδενικού βαθμού (δηλαδή το  $a_0$ ). Για παράδειγμα, το πολυώνυμο  $3^{ου}$  βαθμού έχει τη γενική μορφή:

$$p(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0 \quad (4)$$

και σύμφωνα με το Σχήμα Horner γράφεται ως:

$$p(x) = ((a_3x + a_2)x + a_1)x + a_0 \quad (5)$$

Σκοπός της παρούσας υποεργασίας είναι η ανάπτυξη ενός προγράμματος στη γλώσσα προγραμματισμού C για τον υπολογισμό της τιμής ενός πολυωνύμου και με τους δύο προαναφερθέντες τρόπους, δηλαδή και με υπολογισμό των δυνάμεων του  $x$  και με χρήση του Σχήματος Horner. Ζητείται:

**A.** Να γραφεί συνάρτηση  $dyn$ , η οποία θα δέχεται ως ορίσματα μια διπλής ακρίβειας μεταβλητή  $x$  και μια ακέραια μεταβλητή  $k$  και θα επιστρέφει το αποτέλεσμα  $x^k$ .

**B.** Να γραφεί συνάρτηση  $p$  που θα δέχεται ως ορίσματα έναν δείκτη  $s$  σε πίνακα αριθμών διπλής ακρίβειας, ο οποίος θα περιέχει τις τιμές των συντελεστών του πολυωνύμου, μια διπλής ακρίβειας μεταβλητή  $x$ , για την τιμή της οποίας θα υπολογιστεί η τιμή του πολυωνύμου, και μια ακέραια μεταβλητή  $n$  που θα αντιστοιχεί στο βαθμό του πολυωνύμου. Η συνάρτηση θα επιστρέφει την τιμή του πολυωνύμου σύμφωνα με την εξίσωση (2). Η συνάρτηση  $p$  θα καλεί τη συνάρτηση  $dyn$  για τον υπολογισμό των δυνάμεων της  $x$ .

**Γ.** Να γραφεί συνάρτηση  $h$ , η οποία θα δέχεται ως ορίσματα έναν δείκτη  $s$  σε πίνακα αριθμών διπλής ακρίβειας που θα περιέχει τις τιμές των συντελεστών του πολυωνύμου, μια διπλής ακρίβειας μεταβλητή  $x$ , για την τιμή της οποίας θα υπολογιστεί η τιμή του πολυωνύμου, και μια ακέραια μεταβλητή  $n$  που θα αντιστοιχεί στο βαθμό του πολυωνύμου. Η συνάρτηση θα επιστρέφει την τιμή του πολυωνύμου σύμφωνα με το Σχήμα Horner και την εξίσωση (3).

**Δ.** Να γραφεί κυρίως πρόγραμμα στο οποίο:

- (i) θα δηλώνεται ένας πίνακας  $c$  10 αριθμών διπλής ακρίβειας (στον οποίον θα αποθηκευτούν οι συντελεστές του πολυωνύμου), μια ακέραια μεταβλητή  $b$  (που θα αντιστοιχεί στο βαθμό του πολυωνύμου) και μια μεταβλητή διπλής ακρίβειας  $x$  (που θα αντιστοιχεί στην τιμή για την οποία θέλουμε να υπολογίσουμε την τιμή του πολυωνύμου).
- (ii) θα διαβάζεται με χρήση αμυντικού προγραμματισμού η τιμή της  $b$ , ώστε να διασφαλίζεται ότι θα εισαχθεί τιμή μη αρνητική και μικρότερη του 10.
- (iii) θα διαβάζονται οι  $b+1$  συντελεστές του πολυωνύμου ξεκινώντας από το συντελεστή μηδενικού βαθμού έως το συντελεστή βαθμού  $b$ . Διευκρινίζεται ότι ο συντελεστής μηδενικού βαθμού (δηλαδή ο σταθερός όρος) αποθηκεύεται στην πρώτη θέση του πίνακα  $c$ .
- (iv) θα υπολογίζεται και τυπώνεται η τιμή του πολυωνύμου με χρήση της συνάρτησης  $p$ .
- (v) θα υπολογίζεται και τυπώνεται η τιμή του πολυωνύμου με χρήση της συνάρτησης  $h$ .

#### ΥΠΟΕΡΓΑΣΙΑ 4.

(βαθμοί 20)

Αντικείμενο της παρούσας υποεργασίας είναι ο χειρισμός μίας συλλογής κειμένων που αναπαρίστανται με το μοντέλο διανυσματικού χώρου. Πιο συγκεκριμένα, θέλουμε να αναπαραστήσουμε ένα σύνολο από κείμενα και να εντοπίζουμε την ομοιότητα των κειμένων με ερωτήματα, όπου κάθε ερώτημα είναι και αυτό ένα μικρό κείμενο. Στο μοντέλο διανυσματικού χώρου θεωρείται ότι κάθε κείμενο και κάθε ερώτημα περιγράφονται από ένα σύνολο αντιπροσωπευτικών λέξεων κλειδιών, που ονομάζονται *όροι δεικτοδότησης*. Ένας όρος δεικτοδότησης είναι μια λέξη με σημασιολογικό περιεχόμενο που περιλαμβάνει ένα μέρος του

θέματος με το οποίο ασχολείται το κείμενο. Ο σκοπός του μοντέλου είναι να ορίσει μία κατάλληλη μετρική ομοιότητας μεταξύ των κειμένων και των ερωτημάτων.

Έστω  $N$  ο συνολικός αριθμός των κειμένων και  $D=\{d_0, d_1, \dots, d_{N-1}\}$  τα κείμενα. Θεωρήστε ότι  $size$  είναι το πλήθος των όρων δεικτοδότησης στο σύστημα, ενώ  $t_i$  είναι ένας γενικός όρος δεικτοδότησης, δηλαδή  $T=\{t_0, t_1, \dots, t_{size-1}\}$  είναι το σύνολο όλων των όρων δεικτοδότησης. Συμβολίζουμε με  $freq_{i,j}$  τη συχνότητα εμφάνισης του όρου  $t_i$  στο  $d_j$ , και με  $idf_i$  την αντίστροφη συχνότητα εμφάνισης για τον όρο  $t_i$ , που δίνεται από τον τύπο  $idf_i = \frac{N}{n_i}$ , όπου  $n_i$  το πλήθος

κειμένων που περιέχουν τον όρο  $t_i$  (αν  $n_i=0$ , τότε  $idf_i=0$ ). Για παράδειγμα παρατηρήστε στον Πίνακα 2 τις τιμές  $freq_{i,j}$ ,  $idf_i$  για 5 κείμενα με 4 όρους δεικτοδότησης.

|       | $d_0$ | $d_1$ | $d_2$ | $d_3$ | $d_4$ | $n_i$   | $idf_i$     |
|-------|-------|-------|-------|-------|-------|---------|-------------|
| $t_0$ | 1     | 2     | 2     | 1     | 1     | $n_0=5$ | $idf_0=5/5$ |
| $t_1$ | 1     | 1     | 4     | 0     | 0     | $n_1=3$ | $idf_1=5/3$ |
| $t_2$ | 0     | 1     | 0     | 1     | 1     | $n_2=3$ | $idf_2=5/3$ |
| $t_3$ | 0     | 0     | 0     | 3     | 1     | $n_3=2$ | $idf_3=5/2$ |

**Πίνακας 2**

Τότε ένα από τα πιο γνωστά σχήματα καταγραφής της βαρύτητας συμμετοχής του όρου  $t_i$  στην περιγραφή του περιεχομένου του κειμένου  $d_j$  είναι:

$$w_{i,j} = freq_{i,j} \times idf_i.$$

Για το παράδειγμα του Πίνακα 2 ισχύει:

$$w_{1,2} = freq_{1,2} \times idf_1 = 4 \times (5/3)$$

Αντιστοίχως,  $freq_{i,q}$  είναι η συχνότητα εμφάνισης του όρου  $t_i$  στο κείμενο που αντιπροσωπεύει ένα ερώτημα  $q$ . Για παράδειγμα, προσέξτε τον Πίνακα 3.

|     | $t_0$ | $t_1$ | $t_2$ | $t_3$ |
|-----|-------|-------|-------|-------|
| $q$ | 0     | 1     | 2     | 1     |

**Πίνακας 3**

Τότε για το ερώτημα  $q$  η βαρύτητα συμμετοχής του όρου  $t_i$  δίνεται από τον τύπο:

$$w_{i,q} = freq_{i,q} \times idf_i$$

Για το παράδειγμα του Πίνακα 3 ισχύει:

$$w_{2,q} = freq_{2,q} \times idf_2 = 2 \times (5/3)$$

Στο μοντέλο διανυσματικού χώρου κάθε κείμενο  $d_j$  και κάθε ερώτημα  $q$  αναπαρίστανται ως διανύσματα  $\vec{d}_j$  και  $\vec{q}$  με μία συνιστώσα για κάθε όρο δεικτοδότησης  $t_i$ , όπου οι συνιστώσες είναι τα βάρη  $w_{i,j}$  και  $w_{i,q}$ . Με βάση την αναπαράσταση αυτή ο βαθμός της ομοιότητας μεταξύ του κειμένου  $d_j$  και του ερωτήματος  $q$  υπολογίζεται με τον εξής τύπο:

$$sim(d_j, q) = \frac{\sum_{i=0}^{size-1} (w_{i,j} \times w_{i,q})}{\sqrt{\sum_{i=0}^{size-1} w_{i,j}^2} \times \sqrt{\sum_{i=0}^{size-1} w_{i,q}^2}} \quad (6)$$

Σκοπός της υποεργασίας είναι η μελέτη των δυνατοτήτων της C για την αποτελεσματική υλοποίηση του μοντέλου διανυσματικού χώρου σε ένα σύνολο κειμένων. Θεωρήστε, λοιπόν, έναν πίνακα ακέραιων αριθμών  $F$  (διαστάσεων  $size \times N$  στοιχείων), ο οποίος μοντελοποιεί μία συλλογή κειμένων, και έναν πίνακα ακέραιων αριθμών  $Q$ ,  $size$  στοιχείων που συμβολίζει ένα τυχαίο ερώτημα  $q$ . Ισχύει δηλαδή ότι  $F[i][j] = freq_{i,j}$  και  $Q[i] = freq_{i,q}$ .

Ζητείται να γραφούν οι εξής συναρτήσεις στη γλώσσα προγραμματισμού C (με  $size$  συμβολίζεται το πλήθος των όρων):

**A.** μία συνάρτηση που θα διαβάζει τα στοιχεία του πίνακα  $F$  και τα στοιχεία του πίνακα  $Q$ . Η συνάρτηση να δηλωθεί ως `void read_frequencies(int F[][N], int Q[], int size)`. Θεωρήστε ότι η συνάρτηση πραγματοποιεί τον κατάλληλο αμυντικό προγραμματισμό ώστε οι τιμές που δίνονται και περιγράφουν τις συχνότητες εμφάνισης των όρων  $freq_{i,j}$  και  $freq_{i,q}$  να είναι ακέραιες μεγαλύτερες ή ίσες με το 0.

**B.** μία συνάρτηση που θα δέχεται ως όρισμα τα στοιχεία του πίνακα  $F$  και για κάθε όρο θα υπολογίζει την αντίστροφη συχνότητα εμφάνισης και θα αποθηκεύει τα αποτελέσματα σε ένα πίνακα μίας διαστάσεως με το όνομα  $idf$ , που θα δίνεται ως όρισμα. Η συνάρτηση να δηλωθεί ως `void invert_frequencies(int F[][N], double idf[], int size)`.

**Γ.** μία συνάρτηση που θα δέχεται ως είσοδο τον πίνακα  $F$ , τον πίνακα  $Q$ , τον πίνακα αντιστρόφων συχνοτήτων εμφάνισης  $idf$  και θα αποθηκεύει σε έναν πίνακα μίας διάστασης με το όνομα  $similarity$  την ομοιότητα κάθε κειμένου της συλλογής με το ερώτημα  $q$  με βάση την εξίσωση (6). Στο τέλος η συνάρτηση θα εκτυπώνει τους αριθμούς των κειμένων και την αντίστοιχη τιμή ομοιότητας. Η συνάρτηση να δηλωθεί ως:

`void report_query(int F[][N], int Q[], double idf[], double similarity[N], int size)`.

Ελέγξτε τον κώδικά σας με ένα κυρίως πρόγραμμα που θα δηλώνει τις κατάλληλες μεταβλητές και θα καλεί τις ανωτέρω συναρτήσεις με τη σειρά που ορίστηκαν. Ακολουθεί ένα παράδειγμα εκτέλεσης για  $N=3$  και  $size=4$  (Εικόνα 3).

```

Δώσε τη συχνότητα εμφάνισης του όρου t0 στο κείμενο d0: 0
Δώσε τη συχνότητα εμφάνισης του όρου t0 στο κείμενο d1: 0
Δώσε τη συχνότητα εμφάνισης του όρου t0 στο κείμενο d2: 1
Δώσε τη συχνότητα εμφάνισης του όρου t1 στο κείμενο d0: 0
Δώσε τη συχνότητα εμφάνισης του όρου t1 στο κείμενο d1: 2
Δώσε τη συχνότητα εμφάνισης του όρου t1 στο κείμενο d2: 0
Δώσε τη συχνότητα εμφάνισης του όρου t2 στο κείμενο d0: 1
Δώσε τη συχνότητα εμφάνισης του όρου t2 στο κείμενο d1: 0
Δώσε τη συχνότητα εμφάνισης του όρου t2 στο κείμενο d2: 1
Δώσε τη συχνότητα εμφάνισης του όρου t3 στο κείμενο d0: 0
Δώσε τη συχνότητα εμφάνισης του όρου t3 στο κείμενο d1: 1
Δώσε τη συχνότητα εμφάνισης του όρου t3 στο κείμενο d2: 0
Δώσε τη συχνότητα εμφάνισης του όρου t0 στο ερώτημα q: 0
Δώσε τη συχνότητα εμφάνισης του όρου t1 στο ερώτημα q: 0
Δώσε τη συχνότητα εμφάνισης του όρου t2 στο ερώτημα q: 2
Δώσε τη συχνότητα εμφάνισης του όρου t3 στο ερώτημα q: 0

Ομοιότητες:
Κείμενο d0: 1.000000
Κείμενο d1: 0.000000
Κείμενο d2: 0.447214
  
```

Εικόνα 3

Υπόδειξη:

- για τον υπολογισμό της τετραγωνικής ρίζας χρησιμοποιήστε τη συνάρτηση `double sqrt(double num)` της βιβλιοθήκης `<math.h>`.



**Γενικές Υποδείξεις:****I) Αμυντικός προγραμματισμός και σχολιασμός κώδικα**

Ο κώδικάς σας θα πρέπει να είναι καλά σχολιασμένος και να χρησιμοποιεί στοιχεία αμυντικού προγραμματισμού, όπου αυτό ζητείται. Ειδικά για τον αμυντικό προγραμματισμό, θα πρέπει ο κώδικάς σας να εντοπίζει και να απορρίπτει λανθασμένες εισόδους του χρήστη (εμφανίζοντας το ανάλογο μήνυμα) και να τον προτρέπει να εισάγει αποδεκτή είσοδο. Για την εφαρμογή του αμυντικού προγραμματισμού αρκεί να γίνεται έλεγχος ως προς το εάν μία τιμή που εισάγεται από το χρήστη βρίσκεται εντός αποδεκτών ορίων (π.χ. να είναι θετική, διαφορετική από το 0, μεγαλύτερη από 20 κλπ.). Δεν απαιτείται έλεγχος ως προς το εάν η τιμή ανήκει στο σωστό τύπο δεδομένων (π.χ. ακέραιος, πραγματικός αριθμός, χαρακτήρας κλπ.) σύμφωνα με τον τύπο της μεταβλητής στην οποία θα αποθηκευτεί η τιμή αυτή.

**II) Τρόπος παράδοσης εργασίας**

**α)** Για τις απαντήσεις της εργασίας μπορείτε να ανατρέξετε στη συμπληρωματική βιβλιογραφία που δίνεται και στα βοηθητικά κείμενα που υπάρχουν στο δικτυακό τόπο / portal της θεματικής ενότητας. Συνιστάται να προσθέσετε στο τέλος της εργασίας σας κατάλογο βιβλιογραφίας.

**β)** Στα ερωτήματα που ζητείται υλοποίηση κώδικα σε γλώσσα C, για να θεωρηθούν οι απαντήσεις σας ολοκληρωμένες θα πρέπει:

- Ο κώδικας (όπου ζητείται) να είναι επαρκώς σχολιασμένος και ενσωματωμένος μέσα στο .doc αρχείο του Word, καθώς και σε ξεχωριστό .c αρχείο (ANSI C).
- Στο αρχείο του Word να υπάρχουν εικόνες (screen captures) από την έξοδο των προγραμμάτων.
- Το όνομα κάθε .c αρχείου να περιλαμβάνει το επώνυμό σας με λατινικούς χαρακτήρες, το χαρακτήρα της υπογράμμισης και τον αριθμό του συγκεκριμένου υπο-ερωτήματος (π.χ. αν το επώνυμό σας είναι Γεωργίου, τότε ο κώδικας για την υπο-εργασία 1Γ θα έχει το όνομα Georgiou\_1c.c)
- Κάθε αρχείο C (ANSI C) που θα παραδοθεί θα πρέπει τουλάχιστον να περνά τη φάση της μεταγλώττισης χωρίς λάθη.

**γ)** Για την απάντησή σας θα πρέπει να χρησιμοποιείτε υποχρεωτικά το **Πρότυπο Υποβολής Γραπτής Εργασίας**. Οι απαντήσεις πρέπει να είναι γραμμένες με χρήση **επεξεργαστή κειμένου** (π.χ. **Word**) σε σελίδες **διαστάσεων A4 χωρίς χρώματα**. Το αρχείο της εργασίας θα πρέπει να έχει ως πρώτη σελίδα το Έντυπο Υποβολής - Αξιολόγησης ΓΕ συμπληρωμένο και ως δεύτερη σελίδα τον τίτλο «Σχόλια προς τον φοιτητή» (θα συμπληρωθεί από τον καθηγητή σας). Οι απαντήσεις στις υπο-εργασίες να αρχίζουν από την τρίτη σελίδα, χωρίς να επαναλαμβάνονται οι εκφωνήσεις. Κάθε υπο-εργασία να αρχίζει από νέα σελίδα. Ενδεικτικά, οι απαντήσεις μπορούν να επιτευχθούν σε περίπου 12 σελίδες. Η καλή παρουσίαση της εργασίας λαμβάνεται υπόψη στην αξιολόγηση της εργασίας

**δ)** Τα .c αρχεία με τον πηγαίο κώδικα και το .doc αρχείο κειμένου να υποβληθούν στη διεύθυνση <http://study.eap.gr> όλα μαζί σε συμπίεσμένη μορφή σε ένα αρχείο τύπου .zip, με όνομα αρχείου το επώνυμό σας με λατινικούς χαρακτήρες και τον Αριθμό Μητρώου σας, π.χ. Ioannou\_82345.