



4^ο ΦΥΛΛΑΔΙΟ ΑΣΚΗΣΕΩΝ

ΑΣΚΗΣΗ - 1

1. Να δημιουργηθεί ένα σειριακό αρχείο με όνομα **numbers.txt** στο οποίο θα καταχωρηθούν (με χρήση προγράμματος δημιουργίας σειριακού αρχείου) αριθμοί τύπου float.
2. Να γραφεί πρόγραμμα προσπέλασης του αρχείου **numbers.txt** που θα διαβάζει τους αριθμούς, θα υπολογίζει τον μέσο όρο τους και θα τον καταχωρεί σε ένα νέο σειριακό αρχείο με όνομα **average.txt**. Αν η ανάγνωση του αρχείου **numbers.txt** αποτύχει στο αρχείο **average.txt** θα καταχωρείται ένα κατάλληλο μήνυμα.

ΑΣΚΗΣΗ - 2

Ένα σειριακό αρχείο με όνομα **fortia.txt** περιέχει δεδομένα για τις παραγγελίες προϊόντων μιας βιομηχανίας, που πρέπει να αποσταλούν στους πελάτες. Κάθε παραγγελία, ανάλογα με το είδος του προϊόντος και τις απαιτήσεις του πελάτη αποστέλλεται με διαφορετικό τρόπο. Οι τρόποι αποστολής είναι κωδικοποιημένοι με κωδικούς από 1-10.

Κάθε γραμμή του αρχείου περιέχει πληροφορίες για τον κωδικό της παραγγελίας, την ποσότητα της παραγγελίας σε τεμάχια και τον κωδικό αποστολής (1-10). Όλες οι τιμές είναι θετικοί ακέραιοι αριθμοί. Μια μορφή ορισμένων γραμμών του αρχείου είναι η εξής (οι επικεφαλίδες δεν υπάρχουν στο αρχείο):

ΚΩΔΙΚΟΣ ΠΑΡΑΓΓΕΛΙΑΣ	ΠΟΣΟΤΗΤΑ ΣΕ ΤΕΜΑΧΙΑ	ΚΩΔΙΚΟΣ ΑΠΟΣΤΟΛΗΣ
271	150	4
911	4500	1
123	200	7
....

Ζητούνται να υλοποιηθούν τα παρακάτω :

1. Να γραφεί ο ορισμός μιας κλάσης με όνομα **sale** που θα περιέχει ως private μέλη – δεδομένα τις πληροφορίες που περιγράφονται στο αρχείο.
2. Στη συνάρτηση **main()** να οριστεί ένας πίνακας αντικειμένων της κλάσης **sale** **N** θέσεων (η τιμή του **N** να οριστεί ως σταθερά στην αρχή του προγράμματος).
3. Να γραφεί ένας δομητής (constructor) της κλάσης **sale** για την εισαγωγή δεδομένων στον πίνακα αντικειμένων της κλάσης, **ΜΟΝΟΝ** για εκείνους τους κωδικούς παραγγελίας που έχουν κωδικό αποστολής 1-5. Τα δεδομένα θα διαβάζονται από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου). Αν οι γραμμές του αρχείου τα δεδομένα των οποίων πρέπει να εισαχθούν στον πίνακα αντικειμένων είναι περισσότερες από **N** τότε θα ορίζεται ένας νέος δυναμικός πίνακας ακεραίων αριθμών στον οποίο θα καταχωρείται, για κάθε γραμμή του αρχείου που πρέπει να εισαχθεί στον πίνακα αντικειμένων, **ΜΟΝΟΝ** ο κωδικός παραγγελίας.

4. Χρησιμοποιώντας όλα τα δεδομένα του πίνακα αντικειμένων να δημιουργήσετε δύο νέους δυναμικούς πίνακες που θα περιέχουν ο ένας όλες τις ποσότητες των προϊόντων με κωδικούς αποστολής 1 και 2 και ο άλλος τις ποσότητες των προϊόντων με κωδικούς αποστολής 3,4 και 5. Όταν ολοκληρωθεί η δημιουργία τους το πρόγραμμα θα πρέπει για τον καθένα να εμφανίζει τις τιμές των στοιχείων του καθώς και το άθροισμά τους.

ΑΣΚΗΣΗ – 3

Ένα σειριακό αρχείο με όνομα **machines.txt** περιέχει δεδομένα για τη λειτουργία των εργαλειομηχανών μιας γραμμής παραγωγής. Κάθε γραμμή του αρχείου περιέχει πληροφορίες για τον κωδικό εργαλειομηχανής (υπάρχουν 20 εργαλειομηχανές με κωδικούς 1-20), το πλήθος των τεμαχίων που κατεργάστηκε και το πλήθος των τεμαχίων που μετά την κατεργασία τους απορρίφθηκαν από το τμήμα ποιοτικού ελέγχου (θα αναφέρονται ως ελαττωματικά τεμάχια). Όλες οι τιμές είναι θετικοί ακέραιοι αριθμοί. Ο κωδικός εργαλειομηχανής μπορεί να υπάρχει περισσότερες από μία φορές στο αρχείο. Μια μορφή ορισμένων γραμμών του αρχείου είναι η εξής (οι επικεφαλίδες δεν υπάρχουν στο αρχείο):

ΚΩΔΙΚΟΣ ΕΡΓΑΛΕΙΟΜΗΧΑΝΗΣ	ΠΛΗΘΟΣ ΤΕΜΑΧΙΩΝ	ΠΛΗΘΟΣ ΕΛΑΤΤΩΜ. ΤΕΜΧ.
5	45	7
11	7500	133
14	27	4
11	1500	22
.....		

Ζητούνται να υλοποιηθούν τα παρακάτω :

1. Να γραφεί συνάρτηση δημιουργίας του αρχείου **machines.txt** με χρήση της συνάρτησης δημιουργίας τυχαίων αριθμών για την παραγωγή των δεδομένων.
2. Να γραφεί συνάρτηση εμφάνισης των περιεχομένων του αρχείου στην οθόνη.
3. Να γραφεί ο ορισμός μιας κλάσης με όνομα **machine** που θα περιέχει ως private μέλη – δεδομένα τις πληροφορίες που περιγράφονται στο αρχείο.
4. Στη συνάρτηση **main()** να οριστεί ένας πίνακας αντικειμένων της κλάσης **machine** **N** θέσεων (η τιμή του **N** να οριστεί ως σταθερά στην αρχή του προγράμματος).
5. Να γραφεί ένας δομητής (constructor) της κλάσης **machine** για την εισαγωγή δεδομένων στον πίνακα αντικειμένων της κλάσης. Τα δεδομένα θα διαβάζονται από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου) και θα εισάγονται στον πίνακα αντικειμένων **MONON AN** το ποσοστό των ελαττωματικών τεμαχίων της τρέχουσας γραμμής του αρχείου υπερβαίνει το 10% του πλήθους τεμαχίων. Αν οι γραμμές του αρχείου τα δεδομένα των οποίων πρέπει να εισαχθούν στον πίνακα αντικειμένων είναι περισσότερες από **N** τότε θα ορίζεται ένας νέος δυναμικός πίνακας ακεραίων αριθμών στον οποίο θα καταχωρείται, για κάθε γραμμή που δεν θα μπορέσει να εισαχθεί στον πίνακα αντικειμένων, **MONON** ο κωδικός εργαλειομηχανής.
6. Χρησιμοποιώντας όλα τα δεδομένα του πίνακα αντικειμένων να δημιουργήσετε δύο νέους δυναμικούς πίνακες που θα περιέχουν ο πρώτος το πλήθος ελαττωματικών τεμαχίων που αφορούν τους κωδικούς εργαλειομηχανών 1-10 και ο δεύτερος το πλήθος ελαττωματικών τεμαχίων που αφορούν τους κωδικούς εργαλειομηχανών 11 -20. Όταν ολοκληρωθεί η δημιουργία τους το πρόγραμμα θα πρέπει για τον καθένα να εμφανίζει τις τιμές των στοιχείων του καθώς και το άθροισμά τους.

ΑΣΚΗΣΗ – 4

Ένα κέντρο παραγωγής και διανομής ενέργειας διαβαθμίζει τις περιοχές ευθύνης του με 10 διαφορετικούς κωδικούς (0-9) και χορηγεί ηλεκτρονικές κάρτες πρόσβασης σε όσους θέλουν να αποκτήσουν πρόσβαση (εργαζόμενοι, επισκέπτες κλπ), για οποιονδήποτε λόγο, σε αυτές τις περιοχές.

Ένα σειριακό αρχείο με όνομα **access.txt** καταγράφει τα δεδομένα για τους κωδικούς πρόσβασης όσων επιχειρούν πρόσβαση σε κάποια περιοχή. Για λόγους ασφαλείας η πρόσβαση μπορεί να εκχωρείται ή να αφαιρείται σε οποιονδήποτε, οποιαδήποτε στιγμή.

Κάθε γραμμή του αρχείου περιέχει πληροφορίες για τον κωδικό του ατόμου που επιχειρεί πρόσβαση (τριψήφιος ακέραιος με τιμές από 1-999), τον κωδικό της περιοχής (0-9) και το αν η πρόσβαση (status) ήταν επιτυχής (0) ή όχι (1). Ο κωδικός ενός ατόμου μπορεί να υπάρχει περισσότερες από μία φορές στο αρχείο. Μια μορφή ορισμένων γραμμών του αρχείου είναι η εξής (οι επικεφαλίδες δεν υπάρχουν στο αρχείο):

ΚΩΔΙΚΟΣ ΑΤΟΜΟΥ	ΚΩΔΙΚΟΣ ΠΕΡΙΟΧΗΣ	STATUS
5	5	0
11	7	1
14	0	0
5	9	1
11	1	1
.....		

Ζητούνται να υλοποιηθούν τα παρακάτω :

1. Να γραφεί ο ορισμός μιας κλάσης με όνομα `control` που θα περιέχει ως private μέλη – δεδομένα :
 - a. Τον **κωδικό του ατόμου** (1-999, ωστόσο υπάρχει περίπτωση να υπάρξουν και κωδικοί που δεν έχουν εκχωρηθεί σε άτομα).
 - b. Το **status του ατόμου για όλες τις περιοχές ευθύνης** (δηλ. 10 διαφορετικές τιμές 0 ή 1, μία για κάθε περιοχή ευθύνης). Η αρχική κατάσταση (status) για όλους και για όλες τις περιοχές θα είναι 1.
 - c. Το **πλήθος των προσπαθειών πρόσβασης του ατόμου για όλες τις περιοχές** αντίστοιχα (δηλ. 10 διαφορετικοί μετρητές, ένας για κάθε περιοχή, με αρχική τιμή 0)
2. Στη συνάρτηση `main()` να οριστεί ένας πίνακας αντικειμένων της κλάσης `control` 999 θέσεων. Κάθε θέση του πίνακα αντιστοιχεί σε ένα άτομο, με βάση τον κωδικό του (π.χ. ο κωδικός ατόμου 11 αντιστοιχεί στην 11^η θέση του πίνακα αντικειμένων). Να αρχικοποιηθούν όλες οι θέσεις του πίνακα με χρήση `default constructor`, σύμφωνα με τα παραπάνω.
3. Να γραφεί ένας δομητής (`constructor`) της κλάσης `control` για την εισαγωγή δεδομένων στον πίνακα αντικειμένων της κλάσης. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).
4. Χρησιμοποιώντας όλες τις εγγραφές του αρχείου να ενημερωθούν τα αντίστοιχα μέλη-δεδομένα για κάθε άτομο, με χρήση του `constructor` του ερωτήματος 3 (ΥΠΟΔΕΙΞΗ : το status κάθε ατόμου για μια περιοχή πρόσβασης μπορεί να μεταβάλλεται κατά τη διάρκεια ανάγνωσης του αρχείου).
5. Να δημιουργηθούν δύο νέοι δυναμικοί πίνακες που θα περιέχουν για κάθε άτομο : ο 1^{ος} τον κωδικό του ατόμου και ο 2^{ος} το συνολικό αριθμό προσπαθειών πρόσβασής του για όλες τις περιοχές.

ΑΣΚΗΣΗ – 5

Μια επαναληπτική υπολογιστική διαδικασία επίλυσης ενός προβλήματος διπλασιάζει σε κάθε κύκλο της το πλήθος των επαναλήψεων που εκτελούνται μέχρι να επιτευχθεί συνολικά μια επιθυμητή ακρίβεια, έστω epsilon (η τιμή της ορίζεται ως σταθερά στην αρχή του προγράμματος). Το πρόβλημα αφορά τον προσδιορισμό των τιμών 20 αριθμητικών ποσοτήτων τύπου double (αρχικές τιμές όλες ίσες με 0.0).

Η ακρίβεια κάθε κύκλου επαναλήψεων, έστω pr , είναι το άθροισμα των διαφορών δύο διαδοχικών τιμών και για τις 20 αριθμητικές ποσότητες δηλ. αν x_i^j, x_{i+1}^j είναι δύο διαδοχικές τιμές της ποσότητας $j, j \in [1,20]$ η ακρίβεια είναι:

$$pr = \sum_{j=1}^{j=20} (x_{i+1}^j - x_i^j)$$

Στο τέλος κάθε κύκλου επαναλήψεων αποθηκεύονται οι εξής πληροφορίες:

1. Αύξων αριθμός κύκλου επανάληψης (αρχική τιμή = 1)
2. Οι τιμές των 20 αριθμητικών ποσοτήτων που υπολόγισε ο τρέχων κύκλος (τύπου double)
3. Οι διαφορές των τιμών των 20 αριθμητικών ποσοτήτων από τις τιμές τους στον αμέσως προηγούμενο κύκλο επαναλήψεων (τύπου double)

Να γράψετε τον ορισμό μιας κλάσης με το όνομα calculus με private μέλη-δεδομένα τα παραπάνω.

Η συνάρτηση main() :

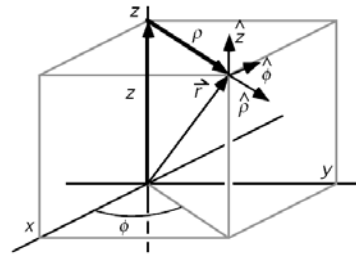
1. Θα δημιουργεί έναν πίνακα αντικειμένων της κλάσης calculus μεγέθους N (η τιμή του N να οριστεί ως σταθερά στην αρχή του προγράμματος). Αμέσως μετά θα πρέπει να καλείται ο default constructor (να δημιουργήσετε τη συνάρτηση του default constructor). Στη συνέχεια θα εισάγει δεδομένα στον πίνακα αντικειμένων **ΜΟΝΟ** με χρήση συνάρτησης constructor **με τους απαραίτητους ελέγχους εγκυρότητας τιμών**. Συνιστάται η χρήση γεννήτριας τυχαίων αριθμών. (ΠΡΟΣΟΧΗ: Η εισαγωγή των δεδομένων θα τερματιστεί αν επιτευχθεί η επιθυμητή ακρίβεια ή αν γεμίσουν οι N θέσεις του πίνακα!).
2. χρησιμοποιώντας όλα τα αντικείμενα του πίνακα στα οποία έχουν εισαχθεί πραγματικά δεδομένα θα βρίσκει και θα εμφανίζει για κάθε μία από τις 20 αριθμητικές ποσότητες τη μέση τιμή τους.
3. χρησιμοποιώντας όλα τα αντικείμενα του πίνακα στα οποία έχουν εισαχθεί πραγματικά δεδομένα θα βρίσκει και θα εμφανίζει σε ποιον κύκλο επαναλήψεων υπάρχει η μέγιστη τιμή pr . Αν η μέγιστη τιμή βρεθεί περισσότερες από μια φορές να δημιουργηθεί ένας νέος δυναμικός πίνακας που θα περιέχει τους α/α αυτών των κύκλων.

ΑΣΚΗΣΗ – 6

Οι σχέσεις που συνδέουν το καρτεσιανό και το κυλινδρικό σύστημα συντεταγμένων είναι:

$$\begin{aligned}\rho &= \sqrt{x^2 + y^2} & x &= \rho \cos\varphi \\ \varphi &= \arctan(y, x) & y &= \rho \sin\varphi \\ z &= z & z &= z\end{aligned}$$

Οι καρτεσιανές συντεταγμένες είναι οι x, y, z και οι κυλινδρικές οι ρ, φ, z .



Ένα σειριακό αρχείο με όνομα **values.txt** περιέχει συντεταγμένες των υλικών σημείων ενός στερεού σώματος. Σε κάθε γραμμή του αρχείου, που αφορά ένα υλικό σημείο, υπάρχουν:

- τρεις αριθμοί τύπου `double` που αντιστοιχούν σε μία από τις δύο προαναφερόμενες τριάδες,
- ένας αριθμός τύπου `double` που αντιστοιχεί στη μάζα m_i του υλικού σημείου
- ένας αριθμός τύπου `int` με τιμές 1 ή 2. Όταν η τιμή είναι 1 οι συντεταγμένες είναι καρτεσιανές, ενώ όταν είναι 2 οι συντεταγμένες είναι κυλινδρικές. Π.χ.

```
3.27 0.93 1.04 2.75 2
1.45 2.678 0.954 2.59 1
.....
```

Να γράψετε μια κλάση με όνομα `m_point` που θα διαχειρίζεται δεδομένα από το αρχείο. Τα μέλη - δεδομένα της κλάσης θα είναι οι καρτεσιανές συντεταγμένες ενός υλικού σημείου (τύπου `double`), οι αντίστοιχες κυλινδρικές (τύπου `double`), και η μάζα (τύπου `double`). Η κλάση πρέπει να περιέχει:

- Έναν `constructor` για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν `constructor` για : την εκχώρηση τιμών από την ανάγνωση του αρχείου και τον υπολογισμό των αντίστοιχων συντεταγμένων.

Στη συνάρτηση `main()`:

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης `m_point` με N θέσεις ($N = \text{γνωστό}$).
2. να αρχικοποιηθούν όλες οι θέσεις του πίνακα με χρήση του κατάλληλου `constructor`, θέτοντας ως εξορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα.
3. να εισαχθούν τιμές από το αρχείο στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου `constructor`. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου) και θα υπολογίζονται οι συντεταγμένες που δεν υπάρχουν στο αρχείο (καρτεσιανές ή κυλινδρικές). Στον πίνακα αντικειμένων θα εισαχθούν τα στοιχεία που αφορούν μόνον τις N πρώτες γραμμές του αρχείου `values.txt`, εάν το αρχείο έχει περισσότερες από N γραμμές.
4. να βρείτε και να εμφανίσετε πόσες γραμμές του αρχείου έχουν καρτεσιανές συντεταγμένες και πόσες κυλινδρικές.
5. να βρείτε και να εμφανίσετε τις καρτεσιανές συντεταγμένες του κέντρου μάζας του στερεού σώματος από όλα τα υλικά σημεία του πίνακα, σύμφωνα με τους παρακάτω τύπους :

$$x_{KM} = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i}, \quad y_{KM} = \frac{\sum_{i=1}^n m_i y_i}{\sum_{i=1}^n m_i}, \quad z_{KM} = \frac{\sum_{i=1}^n m_i z_i}{\sum_{i=1}^n m_i}$$

ΑΣΚΗΣΗ – 7

Θεωρείστε μια εταιρεία που διαχειρίζεται αποθηκευτικούς χώρους. Για κάθε είδος προϊόντων που φθάνουν προς αποθήκευση είναι γνωστά τα παρακάτω στοιχεία:

1. Κωδικός πελάτη (τριψήφιος ακέραιος)
2. κατηγορία προϊόντος (1=ηλεκτρ. συσκευές, 2=μεταλλικά εξαρτήματα, 3=ξυλεία, 4=πλαστικά)
3. ποσότητα (int)
4. χρόνος παραμονής του προϊόντος στις αποθήκες σε ημέρες (int)

Θεωρείστε επίσης ότι σας δίνεται ένα έτοιμο σειριακό αρχείο με όνομα **STORES.TXT** που περιέχει τα προϊόντα που αναμένονται προς αποθήκευση την επόμενη εβδομάδα.

Ζητούνται :

1. Να ορίσετε μια κλάση με όνομα `warepro` που θα έχει ως `private` μέλη τα παραπάνω στοιχεία.
2. Να δημιουργήσετε δύο `random` αρχεία με ονόματα **PRO-A.DAT** και **PRO-B.DAT** που θα περιέχουν τα αντίστοιχα στοιχεία με το σειριακό αρχείο **STORES.TXT** ως εξής :
Το αρχείο **PRO-A.DAT** θα περιέχει τις κατηγορίες προϊόντων 1 και 4 ενώ το αρχείο **PRO-B.DAT** τις κατηγορίες προϊόντων 2 και 3. Κάθε γραμμή του σειριακού αρχείου θα δημιουργεί ένα αντικείμενο της κλάσης `warepro` που θα καταχωρείται στη συνέχεια τυχαία στο αντίστοιχο `random` αρχείο με βάση τον κωδικό πελάτη (δηλ. στη θέση που αντιστοιχεί στον κωδικό πελάτη). Να θεωρείστε ότι η αρχικοποίηση των `random` αρχείων έχει ήδη γίνει και επομένως δεν απαιτείται να υλοποιηθεί.
3. Να υπολογίσετε και να εμφανίσετε τη συνολική ποσότητα προϊόντων που αναμένονται για κάθε κατηγορία προϊόντων και θα παραμείνουν στις αποθήκες περισσότερο από 5 ημέρες.

ΑΣΚΗΣΗ – 8

Μία παραγωγική μονάδα παράγει εξαρτήματα μηχανών. Το σειριακό αρχείο **data.txt** περιέχει πληροφορίες για τα εξαρτήματα αυτά. Κάθε γραμμή του αρχείου αφορά ένα εξάρτημα και περιλαμβάνει:

- κωδικός εξαρτήματος (int) [όρια 1-20]
- ημερομηνία παραγωγής στη μορφή MMEE (μήνας και έτος) με τιμές μήνα 1-12 και έτους 07-16 (δηλ. 2007 – 2016)
- ημερήσιες ποσότητες παραγωγής για κάθε μία από τις 30 ημέρες του μήνα (int). Για ορισμένες από τις ημέρες η παραγωγή μπορεί να είναι μηδενική.

π.χ. (Να θεωρήσετε ότι κάθε κωδικός και για κάθε μήνα κάθε έτους υπάρχει μόνον μία φορά στο αρχείο)

```
14 1114 20 0 14 15 67 0 44 56 0 28 29 .....
7 708 21 11 0 16 0 9 11 0 9 14 7 .....
59 113 7 11 0 0 3 0 4 15 20 0 3 .....
```

Να γραφεί μια κλάση με όνομα **c_parts** που θα διαχειρίζεται τα δεδομένα από το αρχείο **data.txt**. Τα private μέλη – δεδομένα της κλάσης είναι:

- κωδικός εξαρτήματος (int) [όρια 1-20]
- ημερομηνία παραγωγής στη μορφή MMEE, όπως παραπάνω
- πλήθος ποσοτήτων παραγωγής που είναι >0
- ημερήσιες ποσότητες παραγωγής για κάθε μία από τις 30 ημέρες του μήνα (int). Οι θετικές ποσότητες θα καταχωρούνται πρώτες και οι μηδενικές ποσότητες θα καταχωρούνται στο τέλος.

Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).

Στη συνάρτηση main():

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης **c_parts** με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Να θεωρήσετε ως δεδομένο ότι η τιμή του N υπερκαλύπτει το πλήθος των εγγραφών του αρχείου **data.txt**. Στη συνέχεια να γίνει εισαγωγή τιμών στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor.
2. Χρησιμοποιώντας όλα τα αντικείμενα του πίνακα:
 - a. Να βρείτε και να εμφανίσετε τη συνολική ποσότητα παραγωγής κάθε έτους της παραγωγικής μονάδας (δηλ. στα σύνολα θα συμπεριλάβετε όλα τα εξαρτήματα).
 - b. Να δημιουργήσετε και να εμφανίσετε έναν μονοδιάστατο δυναμικό πίνακα που θα περιλαμβάνει, σε μορφή ζευγών, τον κωδικό του εξαρτήματος και τη συνολικά παραχθείσα ποσότητα γι' αυτό το εξάρτημα για καθένα από τα έτη 2007-2016. Ο κωδικός του εξαρτήματος δίνεται στη συνάρτηση main() μέσω της εντολής cin (απαιτείται έλεγχος εγκυρότητας τιμών).

ΑΣΚΗΣΗ – 9

Ένας περιστρεφόμενος άξονας κυκλικής διατομής έχει ακτίνα r ($r =$ γνωστό). Λόγω της χρήσης του παρουσιάζει αλλοιώσεις της γεωμετρίας του σε διάφορα σημεία. Για να προσδιορίσουμε το μέγεθος των αλλοιώσεων πραγματοποιούμε τομές κάθετες στον άξονα περιστροφής και μετράμε τις διαστάσεις 2 διαμέτρων, έστω d_1 και d_2 , κάθετων μεταξύ τους. Οι διαστάσεις των διαμέτρων d_1 και d_2 μαζί με την απόσταση κάθε τομής από την αρχή του άξονα (3 θετικές τιμές τύπου double) καταχωρούνται σε ένα σειριακό αρχείο με όνομα **rotations.txt**. Παράδειγμα ορισμένων τιμών για $r = 100mm$ π.χ.

200.85	199.93	0.4
200.80	199.57	0.55
...		

Να γράψετε τον ορισμό μιας κλάσης με όνομα `ecc` και `private` μέλη-δεδομένα τα παραπάνω. Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).

Στη συνάρτηση `main()`:

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης `ecc` με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξ ορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Να θεωρήσετε ως δεδομένο ότι η τιμή του N υπερκαλύπτει το πλήθος των εγγραφών του αρχείου **rotations.txt**. Στη συνέχεια να γίνει εισαγωγή τιμών στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor.
2. Χρησιμοποιώντας όλα τα στοιχεία του πίνακα αντικειμένων :
 - a. να βρείτε και να εμφανίσετε τη μέση τιμή των αποκλίσεων κάθε μίας από τις διαμέτρους d_1 και d_2 σε σχέση με την αρχική διάμετρο του άξονα. Στον υπολογισμό των μέσων τιμών δεν πρέπει να συμπεριληφθούν οι μηδενικές αποκλίσεις.
 - b. να δημιουργήσετε και να εμφανίσετε ένα δυναμικό πίνακα που θα περιλαμβάνει τις αποστάσεις εκείνων των τομών από την αρχή του άξονα για τις οποίες η απόλυτη τιμή της διαφοράς $d_1 - d_2$ είναι μεγαλύτερη από το 1% της τιμής $2r$.

ΑΣΚΗΣΗ – 10

Οι αριθμοί των πιστωτικών καρτών ακολουθούν μια συγκεκριμένη κωδικοποίηση και αποτελούνται από 16 ψηφία. Ένα σειριακό αρχείο με όνομα `bank_cards.txt` περιέχει δεδομένα πιστωτικών καρτών, σύμφωνα με τα παραπάνω. Κάθε γραμμή του αρχείου περιλαμβάνει τα ψηφία της κάρτας, καθένα ως έναν αριθμό, την ημερομηνία λήξης και το μέγιστο μηνιαίο πιστωτικό όριο π.χ.

```
4 3 8 8 5 7 6 0 1 8 4 1 0 7 0 7    1016    8500.00
4 3 8 8 5 7 6 0 1 8 4 0 2 6 2 6    317     4000.00
```

Να θεωρήσετε ότι το αρχείο περιλαμβάνει στοιχεία πιστωτικών καρτών με αριθμούς που ξεκινούν από 4 (Visa), 5 (Mastercard) και 37 (American Express).

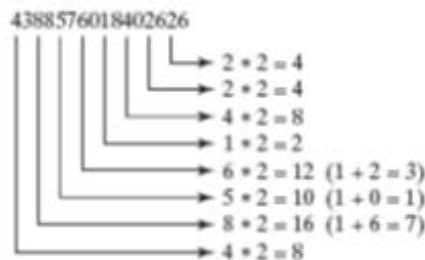
Να γραφεί μια κλάση με όνομα `credit_card` που θα διαχειρίζεται δεδομένα που αφορούν πιστωτικές κάρτες. Τα private μέλη – δεδομένα της κλάσης θα είναι:

- Ένας μονοδιάστατος πίνακας ακεραίων 16 θέσεων για την αποθήκευση των ψηφίων του αριθμού της κάρτας.
- Ένας ακέραιος αριθμός για την ημερομηνία λήξης της κάρτας στη μορφή MMEE (MM = μήνας, EE = έτος)
- Ένας θετικός αριθμός τύπου `double` για το μέγιστο μηνιαίο πιστωτικό όριο της κάρτας

Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).
- Μια συνάρτηση που θα ελέγχει εάν ο αριθμός της κάρτας είναι έγκυρος ή όχι.

Ο αλγόριθμος που χρησιμοποιείται για τον έλεγχο του αριθμού μιας πιστωτικής κάρτας έχει ως στόχο να προσδιορίσει την εγκυρότητα του αριθμού όταν αυτός πληκτρολογείται ή εισάγεται μέσω σάρωσης. Τα βήματα είναι τα εξής :



- Διπλασιασμός κάθε δεύτερου ψηφίου από τα δεξιά προς τα αριστερά. Αν ο διπλασιασμός καταλήγει σε διψήφιο αριθμό τότε τα ψηφία του αριθμού προστίθενται για να προκύψει μονοψήφιος αριθμός.
- Πρόσθεση όλων των ψηφίων από το βήμα 1: $4 + 4 + 8 + 2 + 3 + 1 + 7 + 8 = 37$
- Πρόσθεση όλων των ψηφίων στις περιττές θέσεις, από δεξιά προς τα αριστερά
 $6 + 6 + 0 + 8 + 0 + 7 + 8 + 3 = 38$
- Πρόσθεση των αποτελεσμάτων από τα βήματα 2 και 3: $37 + 38 = 75$
- Αν το αποτέλεσμα από το βήμα 4 διαιρείται με το 10 τότε ο αριθμός της κάρτας είναι έγκυρος.

Στη συνάρτηση main():

6. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης **credit_card** με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξ ορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Να θεωρήσετε ως δεδομένο ότι η τιμή του N υπερκαλύπτει το πλήθος των εγγραφών του αρχείου **bank_cards.txt**. Στη συνέχεια να γίνει εισαγωγή τιμών από το αρχείο στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor, **ΜΟΝΟΝ εάν ο αριθμός της πιστωτικής κάρτας είναι έγκυρος.**
7. Χρησιμοποιώντας όλα τα στοιχεία του πίνακα αντικειμένων :
 - a. Να βρείτε και να εμφανίσετε το πλήθος των πιστωτικών καρτών για κάθε μία από τις Visa, MasterCard και American Express με μέγιστο πιστωτικό όριο μεγαλύτερο από 4999.99
 - b. Να βρείτε και να εμφανίσετε το πλήθος των πιστωτικών καρτών που λήγουν για καθένα από τους επόμενους 15 μήνες, αρχίζοντας από τον Οκτώβριο 2017.

ΑΣΚΗΣΗ – 11

Να γράψετε μια παραλλαγή του παρακάτω προγράμματος που θα επιτρέπει στον χρήστη του να τροποποιήσει (να επικαλύψει) τα στοιχεία μιας συγκεκριμένης εγγραφής εισάγοντας από το πληκτρολόγιο νέα δεδομένα για ένα άτομο. (μπορείτε να χρησιμοποιήσετε τη συνάρτηση `seekp()`).

```
#include <iostream>
#include <fstream> // for file streams
using namespace std;
class person // class of persons
{
protected:
    char name[40]; // person's name
    int age; // person's age
public:
    void showData(void) // display person's data
    {
        cout << "\n Name: " << name;
        cout << "\n Age: " << age;
    }
};

void main(void)
{
    person pers; // create person object
    ifstream infile; // create input file
    infile.open("PERSON.DAT", ios::binary); // open file
    infile.seekg(0, ios::end); // go to 0 bytes from end
    int endposition = infile.tellg(); // find where we are
    int n = endposition / sizeof(person); // number of persons
    cout << "\nThere are " << n << " persons in file";
    cout << "\nEnter person number: "; cin >> n;
    int position = (n-1) * sizeof(person); // number times size
    infile.seekg(position); // bytes from begin
    infile.read( (char*)&pers, sizeof(pers) ); // read one person
    pers.showData(); // display the person
}
```

ΑΣΚΗΣΗ – 12

Μία δοκός στήριξης δέχεται μεταβαλλόμενα αξονικά και στρεπτικά φορτία. Για την εκτίμηση της αντοχής της δοκού γίνονται συνεχώς μετρήσεις του βέλους κάμψης (αξονική φόρτιση) και της γωνίας στρέψης σε διάφορα σημεία της δοκού. Για κάθε μέτρηση, που περιλαμβάνει το βέλος κάμψης και τη γωνία στρέψης είναι γνωστά τα εξής στοιχεία (όλες οι τιμές είναι τύπου double):

- Απόσταση της μέτρησης από το αριστερό άκρο της δοκού σε mm
- Βέλος κάμψης σε mm
- Γωνία στρέψης σε rad
- Αξονικό φορτίο σε kN
- Στρεπτικό φορτίο σε kN·m

Ένα σειριακό αρχείο με όνομα `rod_metrics.txt` περιέχει τις μετρήσεις μιας χρονικής περιόδου, και οι γραμμές του αρχείου είναι:

```
345.2 2.3 0.04 90.5 34.7
```

....

Να γράψετε τον ορισμό μιας κλάσης με όνομα `rod` και private μέλη-δεδομένα τα παραπάνω. Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).

Στη συνάρτηση `main()`:

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης `rod` με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξ ορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Η τιμή του N μπορεί να είναι μικρότερη, μεγαλύτερη ή και ίση με το πλήθος των εγγραφών του αρχείου `rod_metrics.txt`. Στη συνέχεια να γίνει εισαγωγή τιμών στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor.
2. Να βρεθεί (αν υπάρχει) το πλήθος των γραμμών από το αρχείο που δεν περιλαμβάνονται στον πίνακα αντικειμένων.
3. Χρησιμοποιώντας όλα τα στοιχεία του πίνακα αντικειμένων:
 - a. Να θεωρήσετε γνωστά δύο άνω όρια, για το βέλος κάμψης και τη γωνία στρέψης, `limit1` και `limit2` αντίστοιχα. Να δημιουργήσετε έναν δυναμικό πίνακα, και να καταχωρήσετε σε αυτόν, τις αποστάσεις στις οποίες οι τιμές του βέλους κάμψης ΚΑΙ της γωνίας στρέψης, ταυτοχρόνως, υπερβαίνουν αντίστοιχα αυτά τα δύο όρια, μαζί με τις τιμές του βέλους κάμψης και της γωνίας στρέψης, υπό μορφή διαδοχικών τριάδων (δηλ. κάθε καταχώρηση θα περιλαμβάνει μία τριάδα τιμών που είναι η απόσταση, το βέλος κάμψης και η γωνία στρέψης). Στο τέλος να εμφανίσετε τα περιεχόμενα αυτού του δυναμικού πίνακα, μία τριάδα ανά γραμμή.
 - b. Υποθέτοντας ότι η δοκός στήριξης έχει συνολικό μήκος 2000.0 mm, να βρείτε και να εμφανίσετε το μέγιστο και το ελάχιστο βέλος κάμψης για κάθε 200 mm της δοκού (δηλ. για αποστάσεις 0-199mm, 200-399mm, 400-599mm,... από την αρχή της δοκού). Οι τιμές αυτές θα πρέπει να καταχωρούνται σε έναν αριθμητικό πίνακα τύπου `double`, και στη συνέχεια να εμφανίζονται.

ΑΣΚΗΣΗ – 13

Το τμήμα συντήρησης μιας βιομηχανικής μονάδας διατηρεί αρχείο βλαβών για τον μηχανολογικό εξοπλισμό, καταχωρώντας τις εξής πληροφορίες για κάθε εξάρτημα (όλες οι τιμές είναι τύπου int):

- Κωδικός εξαρτήματος (τιμές 1-100)
- Κωδικός μηχανής στην οποία ανήκει το στοιχείο (τιμές 1-20)
- Κατηγορία βλάβης (επιτρεπτές τιμές 1-10)
- Πλήθος ωρών κατά τις οποίες το στοιχείο τέθηκε εκτός λειτουργίας (≥ 0)

Το αρχείο είναι σειριακό, δίνεται, και έχει όνομα **maintain.txt**. Μερικές γραμμές του αρχείου είναι:

```
75    3    6    52
30    2    8    10
...
30    5    1    5
....
```

Κάθε κωδικός εξαρτήματος μπορεί να υπάρχει περισσότερες από μία φορές στο αρχείο και μπορεί να είναι στοιχείο περισσότερων της μίας μηχανών.

Να γράψετε τον ορισμό μιας κλάσης με όνομα **comp** και private μέλη-δεδομένα τα παραπάνω. Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).

Στη συνάρτηση main():

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης **comp** με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξ ορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Η τιμή του N μπορεί να είναι μικρότερη, μεγαλύτερη ή και ίση με το πλήθος των εγγραφών του αρχείου **maintain.txt**. Στη συνέχεια να γίνει εισαγωγή τιμών στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor.
2. Να βρεθεί (αν υπάρχει) το πλήθος των γραμμών από το αρχείο που δεν περιλαμβάνονται στον πίνακα αντικειμένων.
3. Χρησιμοποιώντας όλα τα στοιχεία του πίνακα αντικειμένων:
 - a. Να θεωρήσετε ότι μία μηχανή τίθεται εκτός λειτουργίας όταν η βλάβη σε ένα εξάρτημά της είναι κατηγορίας 8,9 και 10. Να δημιουργήσετε έναν δυναμικό πίνακα που θα περιλαμβάνει διαδοχικά όλες τις μηχανές που τέθηκαν εκτός λειτουργίας από κάθε εξάρτημά τους. Κάθε καταχώρηση στον δυναμικό πίνακα θα περιλαμβάνει τον κωδικό μηχανής (κατά αύξουσα σειρά), ακολουθούμενο από τους κωδικούς των εξαρτημάτων που την έθεσαν εκτός λειτουργίας. Κάθε κωδικός εξαρτήματος θα πρέπει να περιλαμβάνεται μόνον μια φορά για κάθε μηχανή. Οι κωδικοί εξαρτημάτων για κάθε μηχανή δεν απαιτείται να είναι ταξινομημένοι σε αύξουσα σειρά.
 - b. Να βρείτε για κάθε μηχανή το σύνολο των ωρών εκτός λειτουργίας καθώς και το πλήθος των εξαρτημάτων που το δημιούργησαν (ορισμένες ασήμαντες βλάβες δεν προκαλούν διακοπή λειτουργίας της μηχανής, δηλ. έχουν πλήθος ωρών=0).

ΑΣΚΗΣΗ – 14

Η απόσταση δύο σημείων (x_1, y_1, z_1) και (x_2, y_2, z_2) στον τρισδιάστατο χώρο δίνεται από τη σχέση:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

Ένα σειριακό αρχείο με όνομα **data1.txt** περιέχει συντεταγμένες σημείων ενός στερεού σώματος. Σε κάθε γραμμή του αρχείου, που αφορά ένα σημείο, υπάρχουν τρεις αριθμοί τύπου `double` που αντιστοιχούν στις συντεταγμένες του, π.χ.

```
3.27 0.937 1.041
1.45 2.678 0.954
.....
```

Να γράψετε μια κλάση με όνομα `c_point` που θα διαχειρίζεται δεδομένα από το αρχείο. Τα μέλη - δεδομένα της κλάσης θα είναι: ο αύξων αριθμός του σημείου στο αρχείο, οι καρτεσιανές συντεταγμένες ενός σημείου (τύπου `double`), και η απόσταση ενός σημείου από κάποιο άλλο σημείο (τύπου `double`). Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για : την εκχώρηση τιμών από την ανάγνωση του αρχείου και τον υπολογισμό της απόστασης του σημείου από κάποιο άλλο σημείο. Τα χαρακτηριστικά που πρέπει να καλύπτει αυτός ο constructor παρατίθενται στη συνέχεια (ερώτημα 3).

Στη συνάρτηση `main()`:

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης `c_point` με N θέσεις ($N = \text{γνωστό}$).
2. να αρχικοποιηθούν όλες οι θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα.
3. να εισαχθούν τιμές από το αρχείο στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου). Για κάθε 2 διαδοχικές γραμμές του αρχείου θα υπολογίζεται η απόσταση των δύο σημείων, σύμφωνα με τον παραπάνω τύπο, και το 1^ο από τα δύο σημεία κάθε τέτοιου ζεύγους σημείων θα καταχωρείται στον πίνακα αντικειμένων μόνον όταν η απόσταση είναι μεγαλύτερη από 10^{-2} . (ΕΠΕΞΗΓΗΣΗ: θα υπολογίζεται η απόσταση του 1^{ου} σημείου από το 2^ο σημείο στο αρχείο, η απόσταση του 2^{ου} από το 3^ο σημείο στο αρχείο κ.ο.κ.).
4. να βρείτε και να εμφανίσετε τη μέση τιμή των αποστάσεων των σημείων που καταχωρήθηκαν στον πίνακα αντικειμένων από ένα γνωστό σταθερό σημείο με συντεταγμένες (1.0, 2.0, 3.0).

ΑΣΚΗΣΗ – 15

Θεωρείστε ότι σε ένα πάρκο ανεμογεννητριών λειτουργούν 20 ανεμογεννήτριες του ίδιου τύπου και κάθε ανεμογεννήτρια αποτελείται από 10 υποσυστήματα (ηλεκτρικό, υδραυλικό, ελέγχου κλπ.). Για κάθε εξάρτημα που ανήκει σε ένα υποσύστημα μιας ανεμογεννήτριας είναι γνωστά τα εξής:

- κωδικός ανεμογεννήτριας (int) [όρια 1-20]
- κωδικός εξαρτήματος (int) [όρια 1-99]
- κωδικός υποσυστήματος στο οποίο ανήκει (int) [όρια 1-10]
- πλήθος βλαβών του εξαρτήματος για καθένα από τα 10 τελευταία έτη λειτουργίας του πάρκου (10 ακέραιες θετικές τιμές για τα έτη 2006-2015).

Ένα σειριακό αρχείο με όνομα **wind_tb.txt** περιέχει πληροφορίες για τα εξαρτήματα αυτά. Σε κάθε γραμμή του αρχείου οι πληροφορίες αφορούν ένα εξάρτημα, κάθε εξάρτημα υπάρχει μία φορά στο αρχείο για κάθε ανεμογεννήτρια και για κάθε υποσύστημα, π.χ.

```
14 50 2 3 1 5 6 12 4 6 3 8 9
7 77 2 11 14 16 9 9 11 10 9 14 7
19 18 7 1 3 2 3 2 4 1 2 2 3
.....
```

Να γραφεί μια κλάση με όνομα **w_turb** που θα διαχειρίζεται τα δεδομένα από το αρχείο **wind_tb.txt** και private μέλη – δεδομένα τα παραπάνω. Η κλάση πρέπει να περιέχει:

- Έναν constructor για την απόδοση μηδενικών τιμών στα μέλη των αντικειμένων της
- Έναν constructor για την εκχώρηση τιμών από την ανάγνωση του αρχείου. Τα δεδομένα θα διαβάζονται σειριακά από το αρχείο (απαιτείται έλεγχος για την ύπαρξη του αρχείου).

Στη συνάρτηση main() :

1. να δηλώσετε έναν πίνακα αντικειμένων της κλάσης **w_turb** με N θέσεις ($N =$ γνωστό) και να αρχικοποιήσετε όλες τις θέσεις του πίνακα με χρήση του κατάλληλου constructor, θέτοντας ως εξ ορισμού μηδενικές τιμές για όλα τα μέλη - δεδομένα. Να θεωρήσετε ως δεδομένο ότι η τιμή του N υπερκαλύπτει το πλήθος των εγγραφών του αρχείου **wind_tb.txt**. Στη συνέχεια να γίνει εισαγωγή τιμών στα αντικείμενα του πίνακα, με τη χρήση του κατάλληλου constructor.
2. Χρησιμοποιώντας όλα τα αντικείμενα του πίνακα:
 - a. Να βρείτε και να εμφανίσετε το συνολικό πλήθος των βλαβών για κάθε υποσύστημα για όλες τις ανεμογεννήτριες και για τα 10 έτη
 - b. Να δημιουργήσετε και να εμφανίσετε έναν μονοδιάστατο δυναμικό πίνακα που θα περιλαμβάνει, σε μορφή τριάδων, τον κωδικό του εξαρτήματος, τον κωδικό του υποσυστήματος στο οποίο ανήκει και τη μέση τιμή βλαβών για τη δεκαετία για μία ανεμογεννήτρια, ο κωδικός της οποίας δίνεται στη συνάρτηση main() μέσω της εντολής cin (απαιτείται έλεγχος εγκυρότητας τιμών). Η μέση τιμή βλαβών θα είναι στρογγυλοποιημένη στον πλησιέστερο ακέραιο αριθμό.