

ΕΠΛ131 Αρχές Προγραμματισμού

Αναγνώριση: Σημειώνεται πως οι πλείστες από τις πιο κάτω προγραμματιστικές ασκήσεις προέρχονται από το εγχειρίδιο του μαθήματος της Πληροφορικής της Γ' Λυκείου, το οποίο είναι αναρτημένο στην ιστοσελίδα του Υπουργείου Παιδείας και Πολιτισμού. Θερμές ευχαριστίες εκφράζονται επομένως προς τους συγγραφείς του εγχειριδίου αυτού καθώς και όλους όσους συνέβαλαν στην αναβάθμιση του μαθήματος της Πληροφορικής της Γ' Λυκείου. Η αναβάθμιση αυτή (η οποία στηρίζεται στη χρήση της C++) αναμφίβολα συνιστά μια σημαντική εξέλιξη για το αντικείμενο της Πληροφορικής στη δευτεροβάθμια εκπαίδευση, και αλυσιδωτά θα οδηγήσει στην αναθεώρηση του περιεχομένου του μαθήματος ΕΠΛ131 από την ακαδημαϊκή χρονιά 2018/19 με κύρια επίδραση την περαιτέρω ενίσχυση των δύο βασικών πυλώνων του μαθήματος που δεν είναι άλλοι από την αφαιρετικότητα προγράμματος (συναρτήσεις/μέθοδοι και αρθρωτός σχεδιασμός) και την αφαιρετικότητα δεδομένων (νέοι τύποι δεδομένων και αντικείμενα).

Προγραμματιστικές Ασκήσεις

1. Στον ακόλουθο κατάλογο περιλαμβάνονται οι κωδικοί και οι τιμές μονάδας διαφόρων προϊόντων:

| Κωδικός Προϊόντος | Τιμή Μονάδας |
|-------------------|--------------|
| 1 | € 4.00 |
| 2 | € 4.50 |
| 3 | € 5.00 |
| 4 | € 2.00 |
| 5 | € 1.50 |

Να δημιουργήσετε πρόγραμμα, το οποίο λαμβάνει από τη γραμμή εντολής τον κωδικό (code) και την ποσότητα ενός προϊόντος (qty) και τυπώνει το συνολικό ποσό (total), με ακρίβεια δύο δεκαδικών ψηφίων.

2. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής ένα ακέραιο αριθμό N και τυπώνει την ακολουθία των αθροισμάτων από το 1 μέχρι το i, για κάθε i από το 1 μέχρι το N. Για παράδειγμα για N=10, το αποτέλεσμα του προγράμματος είναι το ακόλουθο:

1, 3, 6, 10, 15, 21, 28, 36, 45, 55

3. Να δημιουργήσετε πρόγραμμα το οποίο τυπώνει τον πίνακα προπαίδειας (9 x 9).
4. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής δύο ακέραιους αριθμούς και υπολογίζει το ελάχιστο κοινό πολλαπλάσιο τους.
5. Ένα ασανσέρ έχει ανώτατο όριο βάρους τα 600 κιλά ή τα 8 άτομα. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται το βάρος κάθε ατόμου που μπαίνει, μέχρι που να μην μπορεί να σηκώσει άλλο βάρος ή να φτάσει το όριο των 8 ατόμων, οπότε και θα τυπώσει το συνολικό βάρος και τον αριθμό των ατόμων που έχουν εισέλθει.
6. Να δημιουργήσετε πρόγραμμα το οποίο να γεμίζει ένα μονοδιάστατο πίνακα ακεραίων 100 θέσεων, τοποθετώντας την τιμή 0 στις περιττές θέσεις και την τιμή 1 στις ζυγές θέσεις. Στη συνέχεια να εκτυπώνει όλα τα στοιχεία του πίνακα σε μια γραμμή.

7. Να δημιουργήσετε πρόγραμμα το οποίο να διαβάσει 15 ονόματα και να τα αποθηκεύει σε έναν πίνακα. Στη συνέχεια, να ζητά από τον χρήστη να δώσει έναν αριθμό από το 0 μέχρι το 14 και να επιστρέφει το όνομα που βρίσκεται στη θέση αυτή.
8. Σε ένα διαγωνισμό μαθηματικών λαμβάνουν μέρος 10 μαθητές. Τα ονόματα και οι βαθμοί αποθηκεύονται στους πίνακες *onomata* και *vathmoi*, αντίστοιχα. Να δημιουργήσετε πρόγραμμα το οποίο να τυπώνει τα ονόματα των μαθητών που πέτυχαν βαθμολογία μεγαλύτερη του μέσου όρου.
9. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής τις συντεταγμένες δύο σημείων και εκτυπώνει την απόσταση μεταξύ τους.
10. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει δύο ακέραιους αριθμούς από τη γραμμή εντολής και υπολογίζει το μέσο όρο των μονάδων τους. Τροποποιήστε το πρόγραμμα έτσι ώστε να λαμβάνει ένα απροσδιόριστο πλήθος ακεραίων αριθμών από τη γραμμή εντολής.
11. Σε ένα εστιατόριο κάθε πελάτης πληρώνει επιπλέον 5% της αξίας του λογαριασμού, για φιλανθρωπικούς σκοπούς. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής ένα απροσδιόριστο πλήθος ποσών λογαριασμών και υπολογίζει το ποσό που θα πληρωθεί για φιλανθρωπικούς σκοπούς.
12. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής δύο ακέραιους, θετικούς αριθμούς A και B ($0 < A, B, \leq 2^{40}$) και υπολογίζει το άθροισμα των ψηφίων των δεκάδων τους. Τροποποιήστε το πρόγραμμα έτσι ώστε να υπολογίζει το άθροισμα του υψηλότερου ψηφίου του κάθε αριθμού.
13. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής ένα ακέραιο θετικό αριθμό N που αντιπροσωπεύει το βάρος σε κιλά μιας στοίβας από άμμο, και υπολογίζει το ελάχιστο πλήθος κουβάδων χωρητικότητας 5 κιλών, 3 κιλών και ενός κιλού που χρειάζονται για να μαζέψουν την άμμο.
14. Μία βιβλιοθήκη έχει βιβλία που είναι διαθέσιμα για δανεισμό για μία εβδομάδα, για ένα μήνα και βιβλία που δεν δανείζονται. Το κάθε βιβλίο έχει έναν αριθμό μητρώου που είναι μοναδικός. Ο αριθμός μητρώου είναι ακέραιος. Το γινόμενο των δύο τελευταίων ψηφίων του αριθμού μητρώου καθορίζει τον τύπο δανεισμού του, με βάση τον πιο κάτω πίνακα.

| Γινόμενο δύο τελευταίων ψηφίων αριθμού μητρώου | Είδος δανεισμού |
|--|-----------------|
| < 10 | week |
| ≥ 10 | month |
| 0 | never |

Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται τον αριθμό μητρώου και να τυπώνει το είδος δανεισμού.

15. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται το πλήθος των στοιχείων (N) και τα στοιχεία ενός πίνακα ακεραίων αριθμών και αξιοποιεί μεθόδους για την περιστροφή

δεξιόστροφα (right shift) ή αριστερόστροφα (left shift) κατά x θέσεις των στοιχείων του πίνακα. Το πρόγραμμα λαμβάνει τις πληροφορίες για την περιστροφή (δεξιόστροφα/αριστερόστροφα, αριθμός θέσεων) από τη γραμμή εντολής, και εκτυπώνει τον πίνακα όπως αυτός προκύπτει μετά την περιστροφή. Να δοθούν εναλλακτικοί ορισμοί για τις μεθόδους που κάνουν την περιστροφή, όπου στην μια εκδοχή να μπορεί να γίνει χρήση κάποιου βοηθητικού πίνακα, ενώ στην άλλη εκδοχή να μην επιτρέπεται αυτό

16. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται 2 δεκαδικούς αριθμούς και ένα μαθηματικό τελεστή (+, -, x, /) και να τυπώνει το αποτέλεσμα της πράξης με ακρίβεια 2 δεκαδικών ψηφίων. Σε περίπτωση που η μαθηματική πράξη είναι αδύνατη, να τυπώνεται το μήνυμα "Error".
17. Ο διευθυντής μιας εταιρείας που αντιμετωπίζει σοβαρά οικονομικά προβλήματα αποφάσισε να αποκόπτονται από τον μισθό των υπαλλήλων 2 ποσά. Το πρώτο αφορά την ενίσχυση του αποθεματικού και το δεύτερο αφορά την αποπληρωμή δανείου. Οι αποκοπές γίνονται με βάση τον πιο κάτω πίνακα. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται το μισθό ενός υπαλλήλου και τυπώνει τα ποσά των 2 κρατήσεων και τον καθαρό μισθό του υπαλλήλου.

| Μισθός | Κράτηση 1 | Κράτηση 2 |
|------------|-----------|-----------|
| ≤ 800 | 2% | 2.5% |
| 801 – 1200 | 2.5% | 3% |
| > 1200 | 3% | 4% |

18. Ο δήμος αποφάσισε να κατασκευάσει νέο αμφιθέατρο. Στην πρώτη σειρά θα τοποθετηθούν 50 καθίσματα, ενώ σε κάθε επόμενη σειρά θα προστίθενται 6 καθίσματα. Το κόστος κάθε καθίσματος είναι €40, ενώ ο προϋπολογισμός για τα καθίσματα είναι N ($€2000 < N \leq €50000$). Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται έναν ακέραιο N και να υπολογίζει:
- Το πλήθος των σειρών του αμφιθεάτρου.
 - Τη συνολική χωρητικότητα του αμφιθεάτρου.
 - Το ποσό που περίσσεψε (αν υπάρχει) από την κατασκευή.
19. Ένα μικρό χωριό τροφοδοτείται με νερό από μία δεξαμενή χωρητικότητας 400 κυβικών τόνων. Ο μηχανισμός τροφοδοσίας λειτουργεί ως εξής: Από τα μεσάνυκτα μέχρι τις 7 το πρωί γίνεται μόνο εισροή (είσοδος) νερού στη δεξαμενή, ενώ την υπόλοιπη μέρα γίνεται μόνο εκροή (έξοδος). Σε περίπτωση που η εισροή νερού ξεπεράσει τη διαθέσιμη χωρητικότητα της δεξαμενής, τότε το επιπλέον νερό διανέμεται με ένα ειδικό σύστημα στα χωράφια των κατοίκων του χωριού.

Αν η δεξαμενή στην αρχή είναι άδεια, να δημιουργήσετε πρόγραμμα το οποίο να δέχεται την εισροή και εκροή νερού σε κυβικούς τόνους για 20 μέρες και να τυπώνει:

- Τον αριθμό των κυβικών τόνων που υπάρχουν μέσα στη δεξαμενή μετά το τέλος της 20^{ης} ημέρας.
- Τον αριθμό των κυβικών τόνων νερού που κατέληξαν στα χωράφια λόγω υπερχείλισης.

Σημείωση: Δεν υπάρχει περίπτωση να αδειάσει ποτέ τελείως η δεξαμενή.

20. Αν γνωρίζουμε τον όγκο V ($1 \leq V \leq 10^7$) ενός κουτιού, να δημιουργήσετε πρόγραμμα το οποίο υπολογίζει το πλήθος των πιθανών διαστάσεων του και επίσης να τις τυπώνει.
21. Να δημιουργήσετε πρόγραμμα το οποίο:
- Δέχεται 30 ακέραιους αριθμούς (>99) και τους αποθηκεύει στον πίνακα N
 - Ελέγχει για κάθε αριθμό αν είναι αριθμός *ariks*. Αριθμός *ariks* ορίζεται ο αριθμός που το άθροισμα του πιο σημαντικού (αριστερότερου) και του λιγότερου σημαντικού ψηφίου ισούται με το άθροισμα των υπόλοιπων ψηφίων (π.χ. 121, 1335). Αν είναι αριθμός *ariks*, τότε αποθηκεύεται στον παράλληλο πίνακα P η τιμή *true*, διαφορετικά αποθηκεύεται η τιμή *false*.
 - Τυπώνει τους 2 πίνακες τον έναν δίπλα στον άλλο.
22. Να δημιουργήσετε πρόγραμμα το οποίο:
- Δέχεται τα ονόματα και τη φωτεινότητα N αστεριών (το N δίνεται στη γραμμή εντολής). Η φωτεινότητα είναι ένας δεκαδικός αριθμός μέσα στο διάστημα $[-10.0 \dots 10.0]$. Σε περίπτωση που δοθεί λανθασμένη φωτεινότητα, τυπώνεται σχετικό μήνυμα και ζητά από το χρήστη να τη ξαναδώσει.
 - Τυπώνει τα ονόματα των αστεριών με τη μεγαλύτερη φωτεινότητα. Σε περίπτωση που υπάρχουν περισσότερα από ένα αστέρια με μέγιστη φωτεινότητα, να τυπώνονται όλα.
23. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται N ακέραιους αριθμούς (το πλήθος N δίνεται στη γραμμή εντολής) και στη συνέχεια τυπώνει σε μια γραμμή όλες τις θέσεις του πίνακα στις οποίες εμφανίζεται το ελάχιστο στοιχείο του πίνακα, και στην επόμενη γραμμή όλες τις θέσεις του πίνακα που εμφανίζεται το μέγιστο στοιχείο του πίνακα.
24. Να δημιουργήσετε πρόγραμμα, το οποίο αποθηκεύει στον πίνακα A , 10 χαρακτήρες, οι οποίοι είναι κεφαλαία γράμματα του λατινικού αλφαβήτου, και τυπώνει το πλήθος των εμφανίσεων του κάθε χαρακτήρα, όπου οι χαρακτήρες εμφανίζονται σε αλφαβητική σειρά.
25. Η ζαριά με άθροισμα 7, όταν ρίξουμε 2 ζάρια, είναι η πιο συνηθισμένη ζαριά. Στο παιχνίδι *Lucky Seven* ένας παίκτης προσπαθεί να φέρει όσες περισσότερες, συνεχόμενες ζαριές με άθροισμα 7 μπορεί. Όποιος το κατορθώσει βγαίνει νικητής. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται το πλήθος των ζαριών ενός παίκτη, τις ζαριές που έχει ρίξει και να εμφανίζει στην οθόνη τις περισσότερες, διαδοχικές φορές που έφερε άθροισμα 7.
26. Ο κωδικός μιας κλειδαριάς αποτελείται από N θετικούς ακέραιους αριθμούς ($N > 0$). Για λόγους ασφάλειας είναι κωδικοποιημένος ως εξής: Μια ακολουθία M αριθμών ($2 < M < 100$) κρύβει μέσα της τους αριθμούς του κωδικού. Συγκεκριμένα οι κωδικοί αριθμοί βρίσκονται στις θέσεις που ισαπέχουν από 2 πρώτους αριθμούς, έναν στα αριστερά και ένα στα δεξιά. Για παράδειγμα, αν έχουμε την ακολουθία 4 1 2 4 8 9 5, ο κωδικός αριθμός είναι το 8, καθώς απέχει το ίδιο (2 θέσεις) τόσο από το 2 όσο και από το 5. Αν, όμως, έχουμε την ακολουθία 12 3 100 5 21 9 8 7 6 4, οι κωδικοί αριθμοί είναι οι 100 21 9. Να δημιουργήσετε πρόγραμμα που λαμβάνει το πλήθος M , και μια ακολουθία M θετικών ακεραίων αριθμών και τυπώνει τους κωδικούς αριθμούς της κλειδαριάς. Θεωρείστε ότι θα υπάρχει πάντοτε τουλάχιστο ένας κωδικός αριθμός. Επιπρόσθετα αν ένας κωδικός αριθμός βρίσκεται μεταξύ δύο ή περισσότερων ζευγαριών πρώτων αριθμών, θα τυπωθεί μόνο μία φορά.

27. Το τετράγωνο ενός αριθμού προκύπτει όταν πολλαπλασιάσουμε τον αριθμό με τον εαυτό του. Κάποιοι αριθμοί μπορούν να γραφτούν ως άθροισμα δύο τετραγώνων φυσικών αριθμών. Για παράδειγμα, $5 = 1^2 + 2^2$, και $25 = 0^2 + 5^2 = 3^2 + 4^2$. Στο δεύτερο παράδειγμα βλέπουμε ότι κάποιοι αριθμοί μπορούν να γραφτούν με περισσότερους από έναν τρόπο. Κάποιοι άλλοι αριθμοί, όπως ο αριθμός 12, δεν μπορούν να γραφτούν ως άθροισμα τετραγώνων. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται κάποιους φυσικούς αριθμούς και να βρίσκει με πόσους διαφορετικούς τρόπους ο καθένας από αυτούς μπορεί να γραφτεί ως άθροισμα δύο τετραγώνων αριθμών.
28. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει δύο συμβολοσειρές από τη γραμμή εντολής και τυπώνει τον αριθμό 1 αν ξεκινούν με τον ίδιο χαρακτήρα, διαφορετικά να τυπώνει τον αριθμό 0.
29. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής το όνομα ενός χρώματος και τυπώνει τη λέξη "Primary", σε περίπτωση που το χρώμα είναι red, yellow ή blue, διαφορετικά τυπώνει "Not Primary".
30. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής μια συμβολοσειρά και τυπώνει το κάθε γράμμα της σε διαφορετική γραμμή.
31. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής μια συμβολοσειρά που περιέχει πεζούς και κεφαλαίους χαρακτήρες του λατινικού αλφαβήτου και δημιουργεί μια νέα συμβολοσειρά που περιέχει μόνο τα κεφαλαία γράμματα της συμβολοσειράς που δόθηκε. Σε περίπτωση που δεν βρεθούν κεφαλαία γράμματα, να τυπώνεται το μήνυμα "No capital letters found".
32. Να δημιουργήσετε πρόγραμμα το οποίο δημιουργεί και τυπώνει μια συμβολοσειρά η οποία περιέχει τον πρώτο χαρακτήρα της κάθε συμβολοσειράς που έχει λάβει από τη γραμμή εντολής.
33. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής δύο συμβολοσειρές και τυπώνει τα ακόλουθα σε ξεχωριστές γραμμές:
- (i) Το μήκος της κάθε συμβολοσειράς.
 - (ii) Το αποτέλεσμα της ένωσης των συμβολοσειρών.
 - (iii) Τις δύο συμβολοσειρές, αφού ανταλλάξετε τους πρώτους χαρακτήρες τους.
34. Να δημιουργήσετε πρόγραμμα το οποίο διαβάζει 10 ονόματα και εμφανίζει στην οθόνη το όνομα με τους περισσότερους χαρακτήρες.
35. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια λέξη και ένα χαρακτήρα. Αν ο χαρακτήρας υπάρχει μέσα στη λέξη, να τυπώνεται η θέση της πρώτης εμφάνισής του, διαφορετικά να τυπώνεται ο αριθμός -1.
36. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια λέξη και αντικαθιστά τον χαρακτήρα 'a' (αν υπάρχει μέσα στη λέξη) με το χαρακτήρα 'b'.

37. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια λέξη και μετατοπίζει τους χαρακτήρες της κατά έναν προς τα δεξιά (ο τελευταίος χαρακτήρας θα μεταφερθεί στην πρώτη θέση).
38. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει δύο λέξεις, το μήκος των οποίων είναι τουλάχιστο 3 χαρακτήρες, και δημιουργεί μια τρίτη λέξη που είναι η ένωση των πρώτων τριών χαρακτήρων της πρώτης λέξης με τους τρεις τελευταίους χαρακτήρες της δεύτερης λέξης.
39. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει μια συμβολοσειρά και τυπώνει τον i -οστό χαρακτήρα i φορές. Να θεωρήσετε ότι ο πρώτος χαρακτήρας βρίσκεται στη θέση 1.
40. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται τα ονόματα N φοιτητών και εμφανίζει το πλήθος των ονομάτων που ξεκινούν με το γράμμα N και τελειώνουν με το γράμμα S . Το πρόγραμμα σταματά όταν δοθεί η λέξη "STOP".
41. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια συμβολοσειρά που αποτελείται από πεζούς λατινικούς χαρακτήρες και τυπώνει το χαρακτήρα με τις περισσότερες εμφανίσεις. Σε περίπτωση που υπάρχουν περισσότεροι από ένας χαρακτήρας με μέγιστο πλήθος εμφανίσεων, τυπώνεται αυτός που προηγείται αλφαβητικά.
42. Δίνεται μία μαθηματική έκφραση που περιέχει μέσα παρενθέσεις οι οποίες όμως μπορεί να μην είναι ισορροπημένες. Να δημιουργήσετε πρόγραμμα το οποίο να λαμβάνει τη μαθηματική έκφραση ως συμβολοσειρά και να τυπώνει true σε περίπτωση που οι παρενθέσεις είναι ισορροπημένες, διαφορετικά να τυπώνει false.
43. Μια λέξη ονομάζεται παλίνδρομη όταν μπορεί να διαβαστεί με τον ίδιο τρόπο τόσο από αριστερά όσο και από δεξιά. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια λέξη και τυπώνει true αν η λέξη είναι παλίνδρομη, διαφορετικά τυπώνει false.
44. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται το επίθετο του κάθε μαθητή σε μία τάξη και τυπώνει τα επίθετα που ξεκινούν με το αρχικό γράμμα το οποίο εμφανίζεται τις περισσότερες φορές.
45. Σε κάποιο σύστημα κωδικοποίησης, όπου χρησιμοποιούνται μόνο κεφαλαία λατινικά γράμματα, εφαρμόζονται οι ακόλουθοι μετασχηματισμοί: $A \rightarrow C$, $B \rightarrow D$, $C \rightarrow E$, ..., $Y \rightarrow A$, $Z \rightarrow B$. Για την αποκωδικοποίηση εφαρμόζονται οι αντίστροφοι μετασχηματισμοί. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει μια λέξη και τυπώνει τόσο την κωδικοποίησή της, όσο και την αποκωδικοποίησή της που πρέπει να οδηγεί στην αρχική λέξη.
46. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια πρόταση και μια λέξη και τυπώνει το πλήθος των εμφανίσεων της λέξης μέσα στην πρόταση.

47. Η συμβολοσειρά "AAAAABCCCCCDDDDCC" αποτελείται μόνο από χαρακτήρες του λατινικού αλφαβήτου και το μήκος της είναι 18 χαρακτήρες. Βάσει του μεγέθους της, αν αποθηκευτεί με αυτή τη μορφή, θα χρειαστούν 18 bytes. Η χρήση μνήμης μπορεί να περιοριστεί εφαρμόζοντας την εξής μέθοδο συμπίεσης: όταν υπάρχουν περισσότεροι από έναν συνεχόμενοι ίδιοι χαρακτήρες, αυτοί θα διαγράφονται και θα αντικαθίστανται με έναν αριθμό ο οποίος θα υποδεικνύει το πλήθος τους, ακολουθούμενος από ένα αντίγραφο του χαρακτήρα. Για παράδειγμα, η πιο πάνω συμβολοσειρά θα μετατραπεί σε "5AB6C4D2C", η οποία έχει μήκος μόλις 9 χαρακτήρες. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια συμβολοσειρά και τυπώνει τη συμπιεσμένη συμβολοσειρά.
48. Να δημιουργήσετε πρόγραμμα το οποίο τυπώνει το πλήθος των χαρακτήρων, καθώς και ποιοι είναι οι εν λόγω χαρακτήρες, που εμφανίζονται σε κάθε μια από τις συμβολοσειρές που λαμβάνει από τη γραμμή εντολής. Θεωρείστε ότι η κάθε συμβολοσειρά αποτελείται από τους πεζούς χαρακτήρες του λατινικού αλφαβήτου.
49. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται ένα κωδικό και μια συμβολοσειρά (και τα δύο αποτελούνται από κεφαλαία γράμματα του λατινικού αλφαβήτου) και τυπώνει τις θέσεις των χαρακτήρων του κωδικού που εμφανίζονται μέσα στη συμβολοσειρά. Αν ο κωδικός δεν υπάρχει μέσα στη συμβολοσειρά, να τυπώνεται ο αριθμός -1. Οι χαρακτήρες του κωδικού πρέπει να εμφανίζονται μέσα στη συμβολοσειρά με τη σωστή σειρά, αλλά όχι κατά ανάγκη ο ένας δίπλα από τον άλλο. Για παράδειγμα αν ο κωδικός είναι "STOP" και η συμβολοσειρά είναι "ABDSTKLOTP" το πρόγραμμα θα τυπώσει 3 4 7 9.
50. Στο πρότυπο camelCase γράφουμε όλες τις λέξεις ενωμένες μεταξύ τους. Το πρώτο γράμμα της πρώτης λέξης γράφεται με μικρό (πεζό) χαρακτήρα, ενώ για όλες τις υπόλοιπες το πρώτο γράμμα γράφεται με κεφαλαίο. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται μια συμβολοσειρά διατυπωμένη στο πιο πάνω πρότυπο και να τυπώνει το πλήθος των λέξεων, καθώς και τις λέξεις, που υπάρχουν σε αυτή.
51. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει μια συμβολοσειρά και τυπώνει όλα τα διγράμματα (ακολουθίες δύο χαρακτήρων) καθώς και το πλήθος τους μέσα στη συμβολοσειρά. Η εκτύπωση γίνεται με τη σειρά εμφάνισης των διγραμμάτων μέσα στη συμβολοσειρά.
52. Μια διεύθυνση ηλεκτρονικού ταχυδρομείου (email) στη μορφή username@domain είναι συντακτικά ορθή εάν ικανοποιεί τους ακόλουθους περιορισμούς:
- (i) Ο χαρακτήρας @ υπάρχει μόνο μια φορά.
 - (ii) Επιτρέπονται μόνο οι πεζοί χαρακτήρες, οι τελείες και οι αριθμοί.
 - (iii) Ο πρώτος χαρακτήρας (του username) είναι πάντα πεζό γράμμα.
 - (iv) Στο domain υπάρχει πάντα μια τελεία, που δεν είναι όμως ο τελευταίος χαρακτήρας.
 - (v) Τόσο το username, όσο και το domain πρέπει να αποτελούνται από τουλάχιστο 5 χαρακτήρες.

Να δημιουργήσετε πρόγραμμα που να λαμβάνει ένα email και να τυπώνει "valid" αν είναι συντακτικά ορθό, διαφορετικά να τυπώνει "invalid".

53. Παντόγραμμα καλείται μια φράση (όσο γίνεται πιο σύντομη) η οποία περιλαμβάνει όλα ανεξαιρέτως τα γράμματα του αλφαβήτου. Μια τέτοια είναι η φράση "The quick brown fox jumps over the lazy dog". Να δημιουργήσετε πρόγραμμα το οποίο να υπολογίζει αν μια φράση είναι παντόγραμμα ή όχι.

54. Το DNA αποτελείται από τέσσερα νουκλεοτίδια (αδενίνη, γουανίνη, κυτοσίνη και θυμίνη). Στο συγκεκριμένο πρόβλημα δίνεται μια σειρά DNA, την οποία θα πρέπει να μετατρέψετε σε περιοδική.

Μία συμβολοσειρά μήκους N καλείται περιοδική, με περίοδο p , αν ο i -οστός χαρακτήρας είναι ίσος με τον $(i+p)$ -οστό χαρακτήρα για κάθε χαρακτήρα από τον πρώτο έως τον $(N-p)$ -οστό. Για παράδειγμα, οι συμβολοσειρές "CATCATC", "CATCAT", "ACTAC" και "ACT" είναι όλες περιοδικές με περίοδο 3.

Σας δίνεται μια συμβολοσειρά S και ένας ακέραιος K . Να δημιουργήσετε πρόγραμμα το οποίο να υπολογίζει τον ελάχιστο αριθμό αντικαταστάσεων που είναι απαραίτητες, ώστε η τελική συμβολοσειρά να είναι περιοδική, με περίοδο μικρότερη ή ίση του K . Κάθε αντικατάσταση αποτελείται από την αλλαγή ενός μόνο χαρακτήρα από ένα γράμμα σε ένα οποιοδήποτε άλλο. Θεωρείστε ότι ο αριθμός K δίνεται στη γραμμή εντολής ενώ η DNA συμβολοσειρά S περιέχεται σε ένα αρχείο κειμένου.

55. Να δημιουργήσετε πρόγραμμα το οποίο λαμβάνει από τη γραμμή εντολής μια συμβολοσειρά που να αποτελείται από 3 έως 50 λατινικούς χαρακτήρες. Από την εν λόγω συμβολοσειρά δημιουργεί και εκτυπώνει μια άλλη συμβολοσειρά ως ακολούθως:

- (i) Χωρίζει τη συμβολοσειρά σε δύο τυχαία σημεία, δημιουργώντας τρεις ξεχωριστές συμβολοσειρές.
- (ii) Αντιστρέφει τη σειρά των γραμμάτων σε κάθε μια από αυτές τις τρεις συμβολοσειρές.
- (iii) Στο τέλος, ξαναενώνει τις τρεις συμβολοσειρές στην ίδια σειρά με την οποία ήταν αρχικά, καταλήγοντας έτσι στο τελικό αποτέλεσμα.

56. «Συμμετρική» συμβολοσειρά καλείται μια συμβολοσειρά S , της οποίας ο κάθε χαρακτήρας εμφανίζεται τόσες φορές όσες και ο κάθε ένας από τους υπόλοιπους χαρακτήρες. Για παράδειγμα, η συμβολοσειρά "xhgyzz" είναι συμμετρική, ενώ η συμβολοσειρά "aabbcdd" δεν είναι. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται μια συμβολοσειρά και εμφανίζει τη λέξη "YES", αν η συμβολοσειρά είναι συμμετρική, ή τη λέξη "NO", αν δεν είναι. Σε περίπτωση που η συμβολοσειρά δεν είναι συμμετρική, το πρόγραμμα εντοπίζει αν η συμβολοσειρά μπορεί να μετατραπεί σε συμμετρική, αφαιρώντας μόνο ένα χαρακτήρα από αυτή. Σε τέτοια περίπτωση, εμφανίζεται ο αριθμός 1, αλλιώς εμφανίζεται ο αριθμός 0. Θεωρείστε ότι όλοι οι χαρακτήρες της συμβολοσειράς είναι πεζοί χαρακτήρες του λατινικού αλφαβήτου.

57. Μια μέθοδος κρυπτογράφησης κειμένου, π.χ. ηλεκτρονικών μηνυμάτων, είναι η ακόλουθη. Ένας γραμματικός χαρακτήρας (δηλαδή πεζά και κεφαλαία γράμματα του λατινικού αλφαβήτου) αντικαθίσταται από τον όποιο χαρακτήρα (στο εκτυπώσιμο υποσύνολο της ακολουθίας ASCII) βρίσκεται X θέσεις από αυτόν, για δεδομένο θετικό ακέραιο X , θεωρώντας ότι υπάρχει κυκλική περιτύλιξη των εν λόγω χαρακτήρων. Οι υπόλοιποι, μη γραμματικοί, χαρακτήρες, π.χ. spaces, newlines, σημεία της στίξης, κτλ., δεν υφίστανται καμία αλλαγή. Να δημιουργήσετε πρόγραμμα το οποίο κωδικοποιεί το περιεχόμενο ενός αρχείου κειμένου βάσει της πιο πάνω μεθόδου, και το αποθηκεύει σε ένα νέο αρχείο. Ο ακέραιος αριθμός X λαμβάνεται από τη γραμμική εντολής.
58. Να δημιουργήσετε πρόγραμμα το οποίο να υπολογίζει την τιμή της εξίσωσης $x = y! + z!$ αρχικά χωρίς τη χρήση συνάρτησης και στη συνέχεια με τη χρήση συνάρτησης.
59. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα next, η οποία να δέχεται ως παράμετρο ένα ακέραιο αριθμό και να επιστρέφει τον επόμενο του.
60. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα prod, η οποία να δέχεται ως παραμέτρους δύο πραγματικούς αριθμούς και να επιστρέφει το γινόμενό τους, το οποίο να εμφανίζεται στην οθόνη από το κυρίως πρόγραμμα, με δύο δεκαδικά ψηφία.
61. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα caps, η οποία να δέχεται ως παράμετρο ένα χαρακτήρα και να επιστρέφει true, αν ο χαρακτήρας είναι κεφαλαίο γράμμα, αλλιώς να επιστρέφει false.
62. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα mod, η οποία να δέχεται ως παραμέτρους δύο ακέραιους αριθμούς και να επιστρέφει το υπόλοιπο της διαίρεσης του μεγαλύτερου με τον μικρότερο αριθμό. Οι αριθμοί μπορούν να δοθούν με οποιαδήποτε σειρά.
63. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα power, η οποία να δέχεται ως παραμέτρους δύο ακέραιους αριθμούς (x,y) και να επιστρέφει τη δύναμη x^y .
64. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται ένα ακέραιο αριθμό και χρησιμοποιώντας τη συνάρτηση prime να ελέγχει αν ο αριθμός είναι πρώτος ή όχι. Πρώτος αριθμός θεωρείται αυτός που έχει ως διαιρέτες μόνο τον εαυτό του και τον αριθμό ένα, π.χ. οι αριθμοί 2, 3, 5, 11, 17,
65. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται έναν ακέραιο αριθμό (N) και έναν χαρακτήρα (C) και να χρησιμοποιεί τη συνάρτηση triangle, για να τυπώσει ένα ορθογώνιο τρίγωνο που θα έχει N γραμμές και θα σχηματίζεται από το χαρακτήρα C . Για παράδειγμα, για $N=4$ και $C='@'$ θα σχηματιστεί το ακόλουθο ορθογώνιο τρίγωνο:

```
@
@@
@@@
@@@@
```

66. Στην πιο πάνω άσκηση η διεπαφή (ή πρότυπο) της συνάρτησης `triangle` είναι η ακόλουθη:

```
public static void triangle (int N, char C);
```

Επαναδιατυπώστε τη συνάρτηση αυτή, έτσι ώστε η διεπαφή της να είναι η ακόλουθη και τροποποιήστε ανάλογα το κυρίως μέρος του προγράμματος:

```
public static String triangle(int N, char C);
```

67. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα `pyramid`, η οποία να δέχεται ως παράμετρο το ύψος h ($1 < h < 30$) μίας πυραμίδας και να τυπώνει το κατάλληλο (διακριτό) σχήμα, όπως επιδεικνύεται πιο κάτω για $h = 10$.

```
      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
*****
*****
*****
*****
*****
*****
```

Όπως και για την προηγούμενη άσκηση με το ορθογώνιο τρίγωνο, τροποποιήστε τη συνάρτηση `pyramid` (με τις αντίστοιχες τροποποιήσεις στο κυρίως πρόγραμμα) έτσι ώστε η διεπαφή της από

```
public static void pyramid (int h);
```

να γίνει

```
public static String pyramid (int h);
```

68. Τέλειος καλείται ένας ακέραιος αριθμός όταν το άθροισμα των θετικών διαιρετών του, εκτός του ίδιου, είναι ίσο με τον αριθμό. Για παράδειγμα, ο αριθμός 28 είναι τέλειος αριθμός. Οι διαιρέτες του 28 είναι οι 1, 2, 4, 7, 14 και το άθροισμα αυτών είναι ίσο με 28. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί τη συνάρτηση `perfect`, η οποία δέχεται ως παράμετρο έναν ακέραιο θετικό αριθμό και επιστρέφει το μήνυμα "perfect" σε περίπτωση που ο αριθμός είναι τέλειος, ενώ, σε αντίθετη περίπτωση επιστρέφει το μήνυμα "not perfect".

69. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα `positive`, η οποία δέχεται ως παράμετρο ένα πίνακα ακεραίων αριθμών, και επιστρέφει `true` αν όλα τα στοιχεία του πίνακα είναι θετικοί αριθμοί, αλλιώς επιστρέφει `false`. Η καταχώρηση των στοιχείων στο πίνακα θα γίνεται στο κυρίως πρόγραμμα.

70. Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει 5 συμβολοσειρές από το αρχείο in.txt και να χρησιμοποιεί τη συνάρτηση `String reverseString(String st)`, για να αντιστρέψει κάθε μια από τις συμβολοσειρές. Οι αντεστραμμένες συμβολοσειρές να τυπώνονται στο αρχείο out.txt. Τροποποιήστε το πρόγραμμα έτσι ώστε να διαβάζει ένα απροσδιόριστο αριθμό συμβολοσειρών από το αρχείο in.txt.
71. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα `search`, η οποία δέχεται ως παράμετρο μια συμβολοσειρά και ένα χαρακτήρα και επιστρέφει το πλήθος των εμφανίσεων του χαρακτήρα μέσα στη συμβολοσειρά.
72. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα `min3`, η οποία δέχεται ως παραμέτρους τρεις ακέραιους αριθμούς και επιστρέφει το μικρότερο από τους τρεις. Ορίστε εκ νέου τη συνάρτηση `min3` ώστε να κάνει χρήση της συνάρτησης `min2`, η οποία λαμβάνει ως παραμέτρους δύο ακέραιους αριθμούς και επιστρέφει το μικρότερο ανάμεσά τους.
73. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα `palindrome`, η οποία να δέχεται ως παράμετρο μια συμβολοσειρά και να επιστρέφει `true` αν η συμβολοσειρά είναι παλίνδρομη, αλλιώς να επιστρέφει `false`. Παλίνδρομη καλείται η λέξη που διαβάζεται το ίδιο και από δεξιά προς αριστερά, όπως `eve`, `madam`, `deified`.
74. Ο ιδιοκτήτης μιας αλυσίδας τριών εστιατορίων επιθυμεί να του δημιουργήσετε πρόγραμμα το οποίο:
- (i) Να χρησιμοποιεί τη συνάρτηση `FillData`, η οποία δέχεται τις εισπράξεις για κάθε ένα από τα 3 εστιατόρια, για μια εβδομάδα και τις αποθηκεύει στους πίνακες `R1`, `R2` και `R3`. Οι εισπράξεις είναι πραγματικοί αριθμοί που βρίσκονται στο αρχείο in.txt.
 - (ii) Να χρησιμοποιεί τη συνάρτηση `Calculate`, η οποία να υπολογίζει το σύνολο των εισπράξεων για κάθε μέρα και να το αποθηκεύει στον πίνακα `Total`.
 - (iii) Να χρησιμοποιεί τη συνάρτηση `Print`, για να καταχωρήσει τα στοιχεία του πίνακα `Total` μέσα στο αρχείο out.txt.
- Τροποποιήστε το πρόγραμμά σας έτσι ώστε αντί να χρησιμοποιεί τους ξεχωριστούς μονοδιάστατους πίνακες `R1`, `R2`, `R3` και `Total`, να χρησιμοποιεί ένα πίνακα δύο διαστάσεων (7 x 4).
75. Σε ένα μετεωρολογικό σταθμό γίνονται καθημερινά μετρήσεις της θερμοκρασίας. Να δημιουργήσετε πρόγραμμα το οποίο:
- (i) Να δέχεται τις θερμοκρασίες για έναν χρόνο (365 μέρες) και να τις αποθηκεύει στον πίνακα `T`. Οι θερμοκρασίες είναι πραγματικοί αριθμοί και βρίσκονται

αποθηκευμένοι στο αρχείο temp.txt. Η λειτουργία αυτή πρέπει να εκτελείται από τη συνάρτηση Temperature.

- (ii) Να χρησιμοποιεί τη συνάρτηση MaxT, για να καταχωρήσει στο αρχείο max.txt τη μέγιστη θερμοκρασία και τις μέρες που έχει καταγραφεί η μέγιστη θερμοκρασία.
- (iii) Να χρησιμοποιεί τη συνάρτηση Avg5, για να τυπώσει στο αρχείο average.txt τους μέσους όρους των θερμοκρασιών ανά 5 ημέρες. Δηλαδή θα τυπώσει τον μέσο όρο για τις μέρες 1 μέχρι και την 5, τον μέσο όρο για τις μέρες 6 μέχρι και την 10 κλπ.

76. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα sumofdigits, η οποία να δέχεται ως παραμέτρους δύο μονοψήφιους ακέραιους αριθμούς a και b ($1 \leq a, b \leq 9$) και να επιστρέφει το άθροισμα όλων των διψήφιων αριθμών που σχηματίζονται με τα ψηφία από το a μέχρι και το b, συμπεριλαμβανομένων. Για παράδειγμα, η συνάρτηση sumofdigits(1,3) επιστρέφει τον αριθμό 198, δηλαδή το άθροισμα των αριθμών 11, 12, 13, 21, 22, 23, 31, 32, 33.

77. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα permutate, η οποία να δέχεται ως παράμετρο ένα πίνακα χαρακτήρων και να εμφανίζει στην οθόνη όλες τις ακολουθίες χαρακτήρων που δημιουργούνται, αναδιατάσσοντας τους χαρακτήρες του πίνακα με όλους τους πιθανούς τρόπους. Για παράδειγμα, αν τα περιεχόμενα του πίνακα είναι η ακολουθία χαρακτήρων abc, θα εκτυπωθούν οι ακόλουθες διατάξεις χαρακτήρων:

```
bac  
bca  
cba  
cab  
acb  
abc
```

Τροποποιήστε το πρόγραμμά σας έτσι ώστε η διεπαφή της permutate από

```
public static void permutate(char[] st);  
να γίνει  
public static String[] permutate(char[] st);
```

78. Να δημιουργήσετε πρόγραμμα το οποίο να καλεί μια συνάρτηση με το όνομα find, η οποία να δέχεται ως παράμετρο έναν πίνακα ακεραίων και να επιστρέφει το μέγεθος της μέγιστης αύξουσας υπακολουθίας του πίνακα. Όπως δηλώνει το όνομά της, η μέγιστη αύξουσα υπακολουθία είναι η μεγαλύτερη δυνατή υπακολουθία αριθμών του πίνακα σε αύξουσα σειρά, όπου οι αριθμοί δεν είναι απαραίτητα συνεχόμενοι. Για παράδειγμα η μέγιστη αύξουσα υπακολουθία της ακολουθίας 10 23 9 34 27 51 41 62 είναι η 10 23 34 51 62, και επομένως είναι μεγέθους 5.

79. Να δημιουργήσετε τη Βιβλιοθήκη Search, η οποία να περιλαμβάνει μεθόδους σειριακής/γραμμαμικής αναζήτησης, και δυαδικής αναζήτησης ως ακολούθως:

```

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int sequential_search (int[] arr, int target);

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int sequential_search (double[] arr, double target);

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int sequential_search (String[] arr, String target);

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int binary_search (int[] arr, int target);

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int binary_search (double[] arr, double target);

// Επιστρέφει τη θέση του target στον arr αν υπάρχει,
// αλλιώς επιστρέφει -1
public static int binary_search (String[] arr, String target);

```

80. Να δημιουργήσετε τη Βιβλιοθήκη Sort, η οποία να περιλαμβάνει μεθόδους ταξινόμησης, σε αύξουσα σειρά, ως ακολούθως:

```

public static void bubble_sort (int[] arr);

public static void bubble_sort (double[] arr);

public static void bubble_sort (String[] arr);

public static void insert_sort (int[] arr);

public static void insert_sort (double[] arr);

public static void insert_sort (String[] arr);

public static void indexed_sort (int[] arr, int[] index);

public static void indexed_sort (double[] arr, int[] index);

public static void indexed_sort (String[] arr, int[] index);

```

81. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα searchTwo, η οποία λαμβάνει ως παραμέτρους ένα πίνακα χαρακτήρων και ένα στοιχείο (χαρακτήρα) προς αναζήτηση και επιστρέφει το πλήθος των αποκλειστικά δύο διαδοχικών εμφανίσεων στον πίνακα του υπό αναζήτηση χαρακτήρα. Για παράδειγμα, εάν ο πίνακας περιλαμβάνει τους χαρακτήρες ABCDEEEGRREE και ο υπό αναζήτηση χαρακτήρας είναι το E, το πλήθος των αποκλειστικά δύο διαδοχικών εμφανίσεων του είναι 1, αφού η ακολουθία EEE αποτελείται από τρεις και όχι δύο διαδοχικές εμφανίσεις του E.

82. Να δημιουργήσετε πρόγραμμα για τους σκοπούς μιας εταιρείας, το οποίο καλεί μια συνάρτηση με το όνομα getEmail, η οποία λαμβάνει παραμετρικά δύο παράλληλους πίνακες, όπου στον έναν αποθηκεύονται οι αριθμοί μητρώου των υπαλλήλων της εταιρείας, και στον άλλο τα emails τους, καθώς και ένα αριθμό μητρώου, M. Αν ο M υπάρχει στον πίνακα με τους αριθμούς μητρώου, τότε η συνάρτηση τυπώνει το email του, διαφορετικά τυπώνει "Not Found".

83. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα searchAll, η οποία λαμβάνει δύο πίνακες ακεραίων, A και B, και επιστρέφει ένα πίνακα παράλληλο του B, όπου η κάθε είσοδος του περιέχει true αν το αντίστοιχο στοιχείο του B περιλαμβάνεται στον πίνακα A, διαφορετικά περιέχει false. Επομένως, η διεπαφή της συνάρτησης είναι η ακόλουθη:

```
public static boolean[] searchAll(int[] A, int[] B);
```

Το κυρίως πρόγραμμα, βάσει του αποτελέσματος της searchAll, να τυπώνει "All Found", "Some Found", ή "None found".

84. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση δυαδικής αναζήτησης, η οποία να βρίσκει και να επιστρέφει τη θέση της τελευταίας εμφάνισης ενός ακέραίου αριθμού μέσα σε έναν ταξινομημένο πίνακα ακεραίων. Να θεωρήσετε ότι ο ακέραιος προς αναζήτηση θα εμφανίζεται τουλάχιστον μια φορά στον πίνακα. **Σημείωση:** Η συνάρτηση της δυαδικής αναζήτησης που βρίσκει την τελευταία εμφάνιση του υπό αναζήτηση στοιχείου είναι ελαφρώς διαφοροποιημένη από την κλασική εκδοχή της (την οποία έχετε ορίσει στη Βιβλιοθήκη Search) όπου συνήθως γίνεται η παραδοχή ότι κάθε στοιχείο του πίνακα δεν εμφανίζεται πάνω από μία φορά. Σε αυτή την περίπτωση οι δύο εκδοχές της δυαδικής αναζήτησης συγκλίνουν αφού η μια και μοναδική εμφάνιση του στοιχείου ταυτίζεται με την τελευταία (καθώς και την πρώτη) εμφάνισή του.

85. Να δημιουργήσετε πρόγραμμα το οποίο να βρίσκει το πλήθος των εμφανίσεων ενός ακέραίου αριθμού, μέσα σε έναν ταξινομημένο σε αύξουσα σειρά πίνακα ακεραίων. Να θεωρήσετε ότι ο ακέραιος προς αναζήτηση θα εμφανίζεται τουλάχιστον μια φορά στον πίνακα. **Σημείωση:** Να χρησιμοποιήσετε δύο συναρτήσεις δυαδικής αναζήτησης, οι οποίες να βρίσκουν τη θέση της πρώτης και της τελευταίας εμφάνισης του ακέραίου, αντίστοιχα.

86. Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει 100 ακέραιους αριθμούς από το αρχείο inA.txt και να βρίσκει τον αριθμό με τις περισσότερες εμφανίσεις. **Σημείωση:** Χρησιμοποιήστε δύο παράλληλους πίνακες, nums και times, όπου στον πρώτο θα καταχωρούνται οι διαφορετικοί αριθμοί που διαβάζονται από το αρχείο, και στον δεύτερο το πλήθος των εμφανίσεων τους. Οι πίνακες αυτοί να έχουν χωρητικότητα 100 (για να καλύπτεται η περίπτωση όπου και οι 100 αριθμοί είναι διαφορετικοί μεταξύ τους, στην οποία περίπτωση η κάθε είσοδος του times θα έχει την τιμή 1) ενώ το πόσες θυρίδες τους πραγματικά αξιοποιούνται να καταχωρείται σε μια ακέραια μεταβλητή count. Για παράδειγμα, αν και οι 100 αριθμοί είναι οι ίδιοι, τότε η count θα ισούται με 1 και η πρώτη θυρίδα του πίνακα times, που θα είναι και η μοναδική που θα χρησιμοποιηθεί, θα καταλήξει να έχει την τιμή 100.

87. Να δημιουργήσετε πρόγραμμα το οποίο να βρίσκει την τετραγωνική ρίζα ενός ακέραιου θετικού αριθμού N, χωρίς τη χρήση της συνάρτησης sqrt. Το πρόβλημα μπορεί να λυθεί εφαρμόζοντας τον αλγόριθμο της δυαδικής αναζήτησης στο διάστημα [1...N], ως εξής:

Αρχικά θα ορίσουμε: low = 1.0, hi = N, diff = abs(hi-low) και EPS = 0.0000000001. Για όσο diff >= EPS ορίζουμε mid = (low+hi)/2.0 και ελέγχουμε κάθε φορά αν mid*mid > N, οπότεν θα ορίσουμε το hi = mid, αλλιώς θα ορίσουμε το low = mid. Σε κάθε επανάληψη θα υπολογίζουμε ξανά το diff = abs(hi-low). Το αποτέλεσμα να εμφανίζεται με πέντε δεκαδικά ψηφία.

88. Έχουμε μια εξίσωση της μορφής $f(x) = x^2 - bx - c$. Δημιουργήστε ένα πρόγραμμα, το οποίο λαμβάνει από τη γραμμή εντολής τις αριθμητικές τιμές b c L H K, και εκτυπώνει τη μικρότερη τιμή του x, μέσα στο πεδίο τιμών μεταξύ L και H, για την οποία η f(x) θα είναι μεγαλύτερη από τον ακέραιο K.

89. Να δημιουργήσετε ένα πρόγραμμα, το οποίο να δημιουργεί από ένα μονοδιάστατο πίνακα ακεραίων A, ένα δεύτερο πίνακα B, ο οποίος να περιέχει τα στοιχεία του πίνακα A, έχοντας όμως τα μηδενικά στοιχεία μαζεμένα στο τέλος του. Δηλαδή, μόνο τα μηδενικά στοιχεία θα ταξινομούνται και όχι ο υπόλοιπος πίνακας.

90. Να δημιουργήσετε πρόγραμμα, το οποίο καλεί μια συνάρτηση με το όνομα getSingle η οποία λαμβάνει ως παράμετρο ένα πίνακα ακεραίων, ο οποίος περιλαμβάνει μόνο ένα αριθμό που εμφανίζεται μόνο μια φορά, και επιστρέφει αυτόν τον αριθμό, ο οποίος τυπώνεται από το κυρίως πρόγραμμα.

91. Να δημιουργήσετε πρόγραμμα το οποίο να χρησιμοποιεί μία νέα εκδοχή της συνάρτησης ταξινόμησης insert_sort (η οποία να προστεθεί στη Βιβλιοθήκη Sort), για να ταξινομή ένα πίνακα ακεραίων σε αύξουσα σειρά, με βάση το υπόλοιπο της διαίρεσης των στοιχείων του πίνακα με έναν ακέραιο K. Για παράδειγμα, αν στον πίνακα υπάρχουν τα στοιχεία a και b, αν $a\%K < b\%K$ τότε το στοιχείο a θα τοποθετηθεί πριν από το στοιχείο b. Το πρόγραμμα να εμφανίζει τους αριθμούς ταξινομημένους με τον τρόπο του αναφέρεται πιο πάνω. **Σημείωση:** Η διεπαφή της νέας εκδοχής της insert_sort είναι η ακόλουθη:

```
public static void insert_sort(int[] arr, int K);
```

Θα μπορούσατε να προσθέσετε στην Βιβλιοθήκη Sort, αντίστοιχες εναλλακτικές εκδοχές και για τις συναρτήσεις bubble_sort και indexed_sort.

92. Να δημιουργήσετε πρόγραμμα το οποίο δέχεται τα ύψη, ως πραγματικούς αριθμούς, ενός συνόλου ατόμων και υπολογίζει τη μέγιστη διαφορά ύψους μεταξύ δύο ατόμων, την οποία εμφανίζει με δύο δεκαδικά ψηφία.

93. Υπάρχουν T διαφορετικοί τύποι κουτιών. Για κάθε τύπο T υπάρχουν N συγκεκριμένα κουτιά. Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει από το αρχείο weight.txt το βάρος του κάθε κουτιού (πλήθος N) για τον κάθε τύπο κουτιών (T), ως ακέραιο αριθμό. Το πρόγραμμα να παρουσιάζει τα βάρη των κουτιών για τον κάθε τύπο σε φθίνουσα σειρά, και επίσης να δίνει το συνολικό βάρος για τον κάθε τύπο καθώς και για όλους τους τύπους κουτιών. **Σημείωση:** Να γίνει χρήση ενός πίνακα ακεραίων T x N, δηλαδή `int[][] Boxes = new int[T][N]`, και επίσης να γίνει χρήση κάποιας συνάρτησης

ταξινόμησης, καθώς και μιας συνάρτησης για τον υπολογισμό του αθροίσματος των στοιχείων ενός πίνακα ακεραίων.

94. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα `mergeSort`, η οποία λαμβάνει ως παραμέτρους δύο πίνακες ακεραίων, `N` και `M`, και επιστρέφει ένα νέο πίνακα ο οποίος περιλαμβάνει όλα τα στοιχεία των `N` και `M` σε φθίνουσα σειρά. Η διεπαφή της `mergeSort` είναι η ακόλουθη:

```
public static int[] mergeSort( int[] N, int[] M);
```

Για σκοπούς επαναχρησιμοποίησης, μπορείτε να προσθέσετε την πιο πάνω συνάρτηση στη Βιβλιοθήκη `Sort`.

95. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα `bubbleSortOne`, η οποία λαμβάνει ως παράμετρο έναν πίνακα ακεραίων, τον οποίο ταξινομεί σε αύξουσα σειρά πλην των όσων στοιχείων του που ισούνται με τον αριθμό `1`, τα οποία πρέπει να μείνουν αμετακίνητα. Για παράδειγμα, ο πίνακας με τα στοιχεία `1740 532 9000 1 8500 4000 5000 2120 1 8999` ταξινομείται στην ακολουθία `532 1740 2120 1 4000 5000 8500 8999 1 9000`.

Η διεπαφή της `bubbleSortOne` είναι η ακόλουθη (και αυτή η συνάρτηση μπορεί να προστεθεί στη Βιβλιοθήκη `Sort`):

```
public static void bubbleSortOne (int[] arr);
```

96. Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται από τη γραμμή εντολής τουλάχιστο τρία κλάσματα με τη μορφή δύο θετικών ακεραίων, τον αριθμητή και τον παρονομαστή, και να εμφανίζει τα τρία μεγαλύτερα κλάσματα στην ίδια μορφή με τη μορφή εισόδου. Το πρόγραμμα θα εμφανίζει τα 3 μεγαλύτερα κλάσματα σε φθίνουσα σειρά. Σε περίπτωση που δύο κλάσματα είναι ίσα, δίνεται προτεραιότητα σε εκείνο με τον μικρότερο αριθμητή.

97. Να δημιουργήσετε πρόγραμμα το οποίο να διαβάζει από το αρχείο `votes.txt` το όνομα, το επίθετο, και τον αριθμό ψήφων που έλαβε ο κάθε υποψήφιος σε κάποια εκλογή. Το πρόγραμμα να εμφανίζει τους υποψηφίους και τα ποσοστά που εξασφάλισε ο καθένας, πρώτα σε αλφαβητική σειρά του επιθέτου τους, και ύστερα σε φθίνουσα σειρά του ποσοστού που εξασφάλισαν.

98. Να δημιουργήσετε πρόγραμμα το οποίο καλεί μια συνάρτηση με το όνομα `before`, η οποία λαμβάνει ως παραμέτρους δύο πίνακες χαρακτήρων, `A` και `B`, όπου το κάθε στοιχείο του `B` περιέχεται στον `A`, και επιστρέφει ένα πίνακα ακεραίων, παράλληλο ως προς τον `B`, που αναφέρει πόσοι χαρακτήρες στον `A` προηγούνται αλφαβητικά του αντίστοιχου χαρακτήρα του `B`. Επομένως, η διεπαφή της `before` είναι η ακόλουθη:

```
public static int[] before(char[] A, char[] B);
```

99. Στο χορό των αποφοίτων η παράδοση λέει πως τα αγόρια πρέπει να χορέψουν με ένα κοντύτερο κορίτσι και μπορούν να χορέψουν μόνο με ένα κορίτσι κατά τη διάρκεια της βραδιάς. Θεωρώντας ότι το πλήθος των αγοριών και κοριτσιών είναι το ίδιο (`N`), να δημιουργήσετε πρόγραμμα το οποίο να λαμβάνει ως ακεραίους αριθμούς τα ύψη των `N` αγοριών και των `N` κοριτσιών (το πλήθος `N` εισάγεται από τη γραμμή εντολής) και στη

συνέχεια να παρουσιάζει τη βέλτιστη κατάσταση ως προς τα ζεύγη αγοριών και κοριτσιών, δηλαδή την κατάσταση με το μικρότερο αριθμό ατόμων χωρίς ντάμα ή παρτενέρ.

100. Σε ένα δάσος υπάρχουν N δέντρα και M ξυλοκόποι. Ένα δέντρο έχει διάμετρο D και μπορεί να το κόψει ένας ξυλοκόπος με ύψος H μόνο αν $D \leq H$. Ένας ξυλοκόπος μπορεί να κόψει μόνο ένα δέντρο. Αν σας δοθούν ένας πίνακας με τις διαμέτρους των δέντρων και ένας πίνακας με τα ύψη των ξυλοκόπων, να βρείτε αν είναι κατορθωτό να κοπούν όλα τα δέντρα και ποιο είναι το ελάχιστο συνολικό ύψος ξυλοκόπων που απαιτείται, για να γίνει αυτό. Αν δεν είναι δυνατόν να κοπούν όλα τα δέντρα, να εμφανίζεται ο αριθμός -1 , αλλιώς να εμφανίζεται το ελάχιστο συνολικό ύψος που πρέπει να έχουν οι ξυλοκόποι για να κοπούν όλα τα δέντρα.
101. Να δημιουργήσετε πρόγραμμα το οποίο διαβάζει από το αρχείο nums.txt 36 ακέραιους αριθμούς, τους οποίους καταχωρεί σε ένα δισδιάστατο 6×6 πίνακα, A . Διαβάζει επίσης (από το ίδιο αρχείο) 6 επιπρόσθετους ακέραιους αριθμούς τους οποίους καταχωρεί σε ένα μονοδιάστατο πίνακα B . Το πρόγραμμα καλεί τη συνάρτηση `Max` για να υπολογίσει το μέγιστο αριθμό στον πίνακα A , και τη συνάρτηση `equalCol` για να υπολογίσει κατά πόσο ο πίνακας B ταυτίζεται με κάποια από τις στήλες του πίνακα A . Αν ταυτίζεται επιστρέφει τον αριθμό της στήλης, αλλιώς επιστρέφει -1 . Οι διεπαφές των δύο συναρτήσεων είναι οι ακόλουθες:

```
public static int Max (int[6][6] arr);  
public static int equalCol (int[6][6] arr1, int[6] arr2);
```