ΕΡΓΑΣΤΗΡΙΟ 3

Θέμα: Εντολές διακλάδωσης, Bitwise τελεστές

Μορφοποίηση κώδικα

Όσα αναφέραμε στις εκφωνήσεις των lab1, lab2 για τη μορφοποίηση προγραμμάτων ισχύουν κι εδώ. Επιπλέον, στο lab3 ακολουθήστε τις παρακάτω υποδείξεις.

Τοποθέτηση αγκίστρων

Σε εντολές if να χρησιμοποιείτε πάντα άγκιστρα για το σώμα της εντολής (για το else), ακόμη κι αν το σώμα αποτελείται μόνο από μία εντολή.

Στοίχιση

Κάθε φορά που ανοίγετε ένα άγκιστρο {, οι εντολές που βρίσκονται μέσα σε αυτό γράφονται με στοίχιση 4 κενών σε σχέση με την εντολή που χρησιμοποιεί το άγκιστρο (π.χ. if, for, while, switch). Το άγκιστρο που κλείνει } γράφεται στην ίδια στήλη με την αρχή αυτής της εντολής.

Σχόλια

Μη γράφετε σχόλια με ελληνικούς χαρακτήρες γιατί δεν είναι πάντα συμβατοί με το autolab και δε μπορούμε να βαθμολογήσουμε εύκολα τις ασκήσεις σας! Γράψτε αγγλικά ή greeklish.

Μηνύματα εξόδου

Στα μηνύματα εξόδου υπάρχει ένα ακριβώς κενό ανάμεσα σε διαδοχικές λέξεις.

Προσέξτε να εκτυπώνετε το μήνυμα "`##\n`" στα σημεία που ζητείται. Αυτό χρησιμοποιείται από το autolab ώστε να διασπάσει την έξοδο του προγράμματός σας σε τμήματα τα οποία μπορούν να βαθμολογηθούν ανεξάρτητα μεταξύ τους, ώστε να μη χάνετε όλη την άσκηση αν υπάρχει λάθος σε έναν υπολογισμό.

Οδηγίες για την υποβολή στο autolab

1) Αποθηκεύστε το πρόγραμμα για την άσκηση 1 σε αρχείο με όνομα **`lab3.c`**

2) Ακολουθήστε τη διαδικασία σχηματισμού ομάδας για το lab3.

3) Υποβάλετε το αρχείο **`lab3.c`** στο autolab.

ΠΡΟΣΟΧΗ: Έχετε όριο 10 υποβολών. Χρησιμοποιήστε τα αρχεία ελέγχου ώστε να επιβεβαιώσετε ότι το πρόγραμμα σας λειτουργεί σωστά ΠΡΙΝ υποβάλετε στο autolab,

Άσκηση

Σε αυτή την άσκηση θα δείτε ένα παράδειγμα χρήσης δυαδικών (bitwise) τελεστών για την κωδικοποίηση διαφορετικών ειδών δεδομένων σε περιορισμένο αριθμό από bits καθώς και χρήση if και switch.

Δεδομένα

Το πρόγραμμά σας θα πάρει ως είσοδο ένα δεκαεξαδικό αριθμό μεγέθους 32 bits (τύπου unsigned int) στον οποίο αποθηκεύονται κωδικοποιημένες πληροφορίες εισιτηρίου για ένα ποδοσφαιρικό αγώνα της ομάδας Brentford FC.

Η κωδικοποίηση έχει γίνει ως εξής, ξεκινώντας από το λιγότερο σημαντικό bit:

- Στα 7 λιγότερο σημαντικά bits (θέσεις 0-6) έχει αποθηκευτεί ο αριθμός της θέσης.
- Στα επόμενα 5 bits (θέσεις 7-11) έχει αποθηκευτεί η σειρά της θέσης ως "απόσταση" από το γράμμα 'A'. Για παράδειγμα, αν η σειρά είναι 'G' τότε έχει αποθηκευτεί ο αριθμός 6 γιατί το 'G' απέχει 6 γράμματα από το 'A'.
- Στα επόμενα 6 bits (θέσεις 12-17) έχει αποθηκευτεί ένας ακέραιος που αντιστοιχεί στην πύλη του γηπέδου.
- Στα υπόλοιπα bits (θέσεις 18-31) έχει αποθηκευτεί ένας ακέραιος που είναι το άθροισμα των παραπάνω τιμών και χρησιμοποιείται για να ελεγχθεί η εγκυρότητα του εισιτηρίου*.

Παράδειγμα

Τα στοιχεία ενός εισιτηρίου είναι θέση 90, σειρά G, πύλη 18.

Στο δυαδικό σύστημα, το 90 είναι 1011010, η απόσταση (6) του χαρακτήρα 'G' από τον 'A' είναι 00110 και το 18 είναι 010010. Το άθροισμα τους είναι 114 το οποίο στο δυαδικό σύστημα είναι 1110010. Επομένως, το εισιτήριο είναι κωδικοποιημένο ως εξής:

0000 0001 1100 1001 0010 0011 0101 1010

Ο παραπάνω αριθμός στο δεκαεξαδικό σύστημα έχει την τιμή 0x01C9235A. Αυτή θα είναι η είσοδος του προγράμματος.

Ενδιαφέρουσες πληροφορίες: Ο δυαδικός κωδικός του εισιτηρίου περιέχει πληροφορίες για τον αριθμό θέσης, σειρά και πύλη. Το άθροισμα αυτών περιλαμβάνεται επίσης στον κωδικό του εισιτηρίου και λέγεται checksum. Το checksum είναι ένας αριθμός-έλεγχος που προκύπτει από τις υπόλοιπες τιμές ενός κωδικού και προστίθεται στον κωδικό για να επαληθεύουμε ότι τα υπόλοιπα δεδομένα είναι σωστά και δεν έχουν αλλοιωθεί. Δεν είναι πάντα άθροισμα, αλλά πάντα προκύπτει μετά από κάποια μαθηματική επεξεργασία των υπόλοιπων δεδομένων. Ίσως δεν το ξέρατε αλλά το τελευταίο ψηφίο του ΑΦΜ σας είναι ένα checksum!

Λειτουργία προγράμματος

Το πρόγραμμά σας πρέπει να λειτουργεί ως εξής:

1. Εκτυπώνει το μήνυμα "`Ticket code:\n`" και διαβάζει τον κωδικό ενός εισιτηρίου ως δεκαεξαδική τιμή την οποία αποθηκεύει σε μία μεταβλητή τύπου unsigned int.
2. Χρησιμοποιεί υποχρεωτικά μάσκες (ορισμένες στο δεκαεξαδικό σύστημα) και κατάλληλους δυαδικούς (bitwise) τελεστές για να αποσπάσει και να αποθηκεύσει σε ξεχωριστές μεταβλητές τις επιμέρους πληροφορίες που είναι κωδικοποιημένες στον κωδικό του εισιτηρίου. Συστήνουμε να ξεκινήσετε από το λιγότερο σημαντικό bit.
3. Εκτυπώνει μήνυμα στη μορφή "`G R S\n`". Στο μήνυμα, `G` είναι η πύλη , `R` είναι η σειρά ως χαρακτήρας, όπως προκύπτει με βάση την απόσταση από το 'A', και `S` η θέση.
4. Ελέγχει αν το άθροισμα της θέσης, απόστασης και πύλης είναι ίσο με το αντίστοιχο άθροισμα (checksum) που περιέχει ο κωδικός. Αν όχι, εκτυπώνει το μήνυμα "`Invalid ticket.\n`" και τερματίζει άμεσα με χρήση της εντολής `return 1;` (η εντολή αυτή μπορεί να εισαχθεί σε οποιοδήποτε σημείο της main κι έχει ως αποτέλεσμα τον άμεσο τερματισμό της.) Διαφορετικά, εκτυπώνει το μήνυμα "`Valid ticket.\n`" και συνεχίζει κανονικά.
5. Εκτυπώνει το μήνυμα "`##\n`".
6. Εκτυπώνει μήνυμα στη μορφή "`C seat.\n`" όπου `C` το είδος (χρώμα) της θέσης. Οι θέσεις 1-40 είναι "`Red`", οι θέσεις 41-80 είναι "`White`", και οι θέσεις 81-120 είναι "`Yellow`". Για οποιαδήποτε άλλη τιμή το είδος της θέσης είναι "`Standing`".
7. Εκτυπώνει το μήνυμα "`##\n`".
8. Εκτυπώνει το μήνυμα "`Game time:\n`" και διαβάζει την ώρα και λεπτό έναρξης του αγώνα στη μορφή `Ω:Λ` όπου `Ω` η ώρα και `Λ` το λεπτό. Δε χρειάζεται να ελέγξετε αν η ώρα και το λεπτό είναι έγκυρες τιμές .
9. Εκτυπώνει το μήνυμα "`Now:\n`" και διαβάζει την τωρινή ώρα και λεπτό, και πάλι στη μορφή `Ω:Λ`.
10. Εάν η στιγμή έναρξης του αγώνα είναι προγενέστερη της τωρινής, εκτυπώνει το μήνυμα "`Too late.\n`" και τερματίζει άμεσα με χρήση εντολής `return 2;` . Αν είναι ακριβώς ίδια, εκτυπώνει το μήνυμα "`On time.\n`" , διαφορετικά, εκτυπώνει το μήνυμα "`Too early.\n`". Χρησιμοποιήσετε if με σύνθετες συνθήκες για την υλοποίηση αυτού του βήματος (και όχι μετατροπή σε λεπτά). Σκεφτείτε τι είναι πιο εύκολο να ελέγξετε πρώτο.
11. Εκτυπώνει το μήνυμα "`##\n`".
12. Επειδή η συντομογραφία της ομάδας είναι BRE FC, αποφασίστηκε να δοθεί μια έκπτωση σε όσους βρίσκονται σε μια από τις σειρές B, R, E, F, C. Χρησιμοποιήστε εντολή switch (<u>υποχρεωτικά</u>) ώστε το πρόγραμμα να εκτυπώνει:

- το μήνυμα "`30% off.\n`" αν η σειρά είναι 'B'
- το μήνυμα "`25% off.\n`" αν η σειρά είναι 'R'
- το μήνυμα "`20% off.\n`" αν η σειρά είναι 'E'
- το μήνυμα "`10% off.\n`" αν η σειρά είναι είτε 'F' είτε 'C'.
- το μήνυμα "`No sale.\n`" σε κάθε άλλη περίπτωση.

Σημείωση: Οι εκτυπώσεις ακεραίων είναι στο δεκαδικό σύστημα.

Παραδείγματα εκτέλεσης

Σε κίτρινο φόντο εμφανίζεται η είσοδος του χρήστη από το πληκτρολόγιο.

Ticket code: 265e277 30 E 119 Valid ticket. ## Yellow seat. ## Game time: 21:45 Now: 21:10 Too early. ## 20% off.	Ticket code: 8a2000 34 A 0 Valid ticket. ## Standing seat. ## Game time: 17:30 Now: 17:30 On time. ## No sale.
Ticket code: 210c078 12 A 120 Valid ticket. ## Yellow seat. ## Game time: 10:20 Now: 10:45 Too late.	Ticket code: 5123cc8 35 Z 72 Invalid ticket.