

ΣΕΤ ΑΣΚΗΣΕΩΝ 1

ΕΡΓΑΣΤΗΡΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι, ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2013-2014

Προθεσμία: 25/11/13, 22:00

Διαβάστε πριν ξεκινήσετε

Διαβάστε την εκφώνηση προσεκτικά και “σχεδιάστε” το πρόγραμμά σας στο χαρτί.

Για κάθε στάδιο, αποφασίστε τι μεταβλητές θα χρειαστείτε, τι ονόματα θα τους δώσετε, αν χρειάζονται σταθερές κι αν ναι για ποιες ποσότητες, τι δομές ελέγχου θα χρησιμοποιήσετε για κάθε λειτουργία και πώς θα κάνετε τους υπολογισμούς που ζητούνται.

Μη διστάζετε να ζητήσετε βοήθεια! Μπορείτε να χρησιμοποιήσετε κατά προτίμηση το forum προγραμματισμού (<http://inf-server.inf.uth.gr/courses/coding/>) και μόνο αν είναι απαραίτητο email (π.χ. αν πραγματικά επιβάλλεται να στείλετε κάποιο κομμάτι κώδικα μαζί με το μήνυμά σας).

Μην προσπαθήσετε να γράψετε το πρόγραμμα μια κι έξω. Για κάθε στάδιο που ολοκληρώνετε, βεβαιωθείτε ότι κάνει compile και λειτουργεί σωστά και μετά να προχωράτε στο επόμενο στάδιο. Κάθε φορά που τελειώνετε ένα στάδιο, να αποθηκεύετε ένα αντίγραφο του προγράμματος όπως είναι μέχρι εκείνο το σημείο. Έτσι, αν σε κάποιο επόμενο στάδιο καταστραφεί το αρχείο σας, θα έχετε μια παλιότερη έκδοση από την οποία μπορείτε να συνεχίσετε. Τις επιμέρους “εκδόσεις” να τις ονομάζετε με διαφορετικά ονόματα (πχ. hw1stadio1.c, hw1stadio2.c). Εννοείται πως θα παραδώσετε μόνο την τελική έκδοση του προγράμματος.

Σε κάθε στάδιο σας δίνουμε μια εκτίμηση της δυσκολίας του (διαβάθμιση από 1 έως 3 αστεράκια).

Η εργασία αυτή μπορεί να γίνει σε ομάδες μέχρι 2 ατόμων. Δε χρειάζεται να είστε ομάδα με το ίδιο άτομο που είστε στο εργαστήριο. Μπορείτε να συζητάτε τις ασκήσεις με συμφοιτητές σας αλλά δεν επιτρέπεται η ανταλλαγή κώδικα με οποιοδήποτε τρόπο.

Ξεκινήστε νωρίς! Ο προγραμματισμός είναι πάντα ΠΟΛΥ πιο χρονοβόρος από ότι περιμένετε.

Εκπρόθεσμες ασκήσεις δε γίνονται δεκτές.

Οι ασκήσεις σας θα βαθμολογηθούν στα παρακάτω (χωρίς ιδιαίτερη σειρά):

- Ορθότητα
- Γενική μορφοποίηση προγράμματος (στοίχιση, ονόματα μεταβλητών, κτλ.)
- Σχεδιασμός προγράμματος και αποτελεσματική χρήση κατάλληλων δομών, μεταβλητών, σταθερών κτλ.
- Συμμόρφωση με τις προδιαγραφές
- Αποτελεσματικά σχόλια, σύμφωνα με τους κανόνες σχολιασμού του σχετικού φυλλαδίου.

Άσκηση : Διαχείριση Ταμείου σε Χώρο Στάθμευσης Οχημάτων

Εισαγωγή

Θα γράψετε ένα πρόγραμμα το οποίο χρησιμοποιείται σε ένα χώρο στάθμευσης οχημάτων για να διαχειριστεί τις πληρωμές και να καταγράψει στατιστικά στοιχεία. Το πρόγραμμα ανιχνεύει τον τύπο κάθε οχήματος που μπαίνει στο χώρο, υπολογίζει το ποσό που θα πληρωθεί κατά την έξοδο με βάση τη διάρκεια στάθμευσης, διαχειρίζεται τη συναλλαγή και στο τέλος της ημέρας καταγράφει στατιστικά στοιχεία.

Ακολουθούν λεπτομερείς οδηγίες για το πώς πρέπει να λειτουργεί το πρόγραμμά σας και στάδια κατασκευής του. ΜΗΝ προσπαθήσετε να γράψετε όλο το πρόγραμμα σε ένα βήμα γιατί θα κάνετε λάθη και θα σας πάρει πολύ περισσότερο χρόνο.

Αποθηκεύστε το πρόγραμμά σας σε ένα αρχείο με όνομα `hw1.c`

Στάδιο 1: Ανίχνευση τύπου οχήματος (*)

Γράψτε ένα πρόγραμμα το οποίο διαβάζει από το πληκτρολόγιο τον τύπο κάθε οχήματος που μπαίνει στο χώρο στάθμευσης και μετρά πόσα οχήματα κάθε τύπου έχουν σταθμεύσει μέχρι το τέλος της ημέρας. Για την ακρίβεια, το πρόγραμμά σας πρέπει να εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα

Enter vehicle type (M,m,C,c,T,t,B,b) or x to exit:

ακολουθούμενο από ένα κενό. Οι πιθανοί τύποι οχημάτων είναι μοτοσυκλέτα (m ή M), αυτοκίνητο (c ή C), φορτηγό (t ή T) και λεωφορείο (b ή B). Αν δοθεί x θεωρούμε ότι είναι το τέλος της ημέρας και δε θα έρθουν άλλα οχήματα. Αν δοθεί οποιαδήποτε άλλη επιλογή, το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα λάθους

Error: Invalid vehicle type "X"!

όπου X είναι ο άκυρος τύπος οχήματος που δόθηκε, κι επαναλαμβάνει τα παραπάνω βήματα.

Στο τέλος της ημέρας, το πρόγραμμα εκτυπώνει τα παρακάτω:

Τον τίτλο **MOTORCYCLES:** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής

Ένα χαρακτήρα tab, το μήνυμα **Number:X** όπου X το πλήθος μοτοσυκλετών που στάθμευσαν στο χώρο, και χαρακτήρα αλλαγής γραμμής. Ο αριθμός X πρέπει να καταλαμβάνει 3 θέσεις.

Τον τίτλο **CARS:** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής

Ένα χαρακτήρα tab, το μήνυμα **Number:X** όπου X το πλήθος αυτοκινήτων που στάθμευσαν στο χώρο, και χαρακτήρα αλλαγής γραμμής. Ο αριθμός X πρέπει να καταλαμβάνει 3 θέσεις.

Τον τίτλο **TRUCKS:** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής

Ένα χαρακτήρα tab, το μήνυμα **"Number:X"** όπου X το πλήθος φορτηγών που στάθμευσαν στο χώρο, και χαρακτήρα αλλαγής γραμμής. Ο αριθμός X πρέπει να καταλαμβάνει 3 θέσεις.

Τον τίτλο **BUSES:** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής

Ένα χαρακτήρα tab, το μήνυμα **"Number:X"** όπου X το πλήθος λεωφορείων που στάθμευσαν στο χώρο, και χαρακτήρα αλλαγής γραμμής. Ο αριθμός X πρέπει να καταλαμβάνει 3 θέσεις.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Στάδιο 2: Υπολογισμός διάρκειας στάθμευσης (*)**

Προσθέστε κώδικα στο πρόγραμμα που γράψατε για το στάδιο 1 έτσι ώστε να καταγράφονται οι ώρες εισόδου κι εξόδου του κάθε οχήματος και να υπολογίζεται το ποσό που πρέπει να πληρώσουν.

Για την ακρίβεια, όταν δοθεί έγκυρος τύπος οχήματος, το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα

Enter entry time (HH:MM, 24hr clock):

ακολουθούμενο από ένα κενό. Η ώρα εισόδου πρέπει να δοθεί στη μορφή HH:MM (για παράδειγμα 13:20 ή 9:00). Η ώρα πρέπει να είναι μεταξύ 0 και 23 (συμπεριλαμβανομένων) ενώ τα λεπτά μεταξύ 0 και 59 (συμπεριλαμβανομένων). Αν δε δοθούν έγκυρα νούμερα, τότε το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα λάθους

Error: Hour must be between 0 and 23 and minute must be between 0 and 59!

ακολουθούμενο από χαρακτήρα αλλαγής γραμμής κι επαναλαμβάνει τα παραπάνω βήματα.

Εφόσον δοθεί σωστά η ώρα εισόδου, το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα

Enter exit time (HH:MM, 24hr clock):

ακολουθούμενο από ένα κενό και κάνει τον ίδιο έλεγχο εγκυρότητας. Επιπλέον, το πρόγραμμα ελέγχει αν η ώρα εξόδου είναι μεταγενέστερη της ώρας εισόδου (θεωρείστε ότι η είσοδος και η έξοδος γίνονται πάντα την ίδια μέρα). Αν όχι, τότε εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα

Error: Exit time must be after entrance time!

ακολουθούμενο από χαρακτήρα αλλαγής γραμμής κι επαναλαμβάνει το βήμα εισαγωγής της ώρας εξόδου.

Εφόσον έχουν δοθεί σωστά οι ώρες εισόδου κι εξόδου, το πρόγραμμα υπολογίζει πόσο πρέπει να πληρώσει ο οδηγός, ανάλογα με τον τύπο του οχήματος. Οι τιμές ανά ώρα στάθμευσης είναι **0.75** ευρώ για μοτοσυκλέτες, **2.25** για αυτοκίνητα, **5.40** για φορτηγά, **8.60** για λεωφορεία και το σύνολο υπολογίζεται ανά λεπτό.

Στο τέλος της ημέρας, το πρόγραμμα εκτυπώνει τα παρακάτω:

Στην κατηγορία MOTORCYCLES, μετά το μήνυμα Number:X που γράψατε στο προηγούμενο στάδιο, εκτυπώνεται ένας χαρακτήρας tab, το μήνυμα **Average parking time: H hrs M min** όπου H και M είναι η μέση διάρκεια στάθμευσης μοτοσυκλετών σε ώρες και λεπτά αντίστοιχα, και χαρακτήρας αλλαγής γραμμής. Τόσο το H, όσο και το M καταλαμβάνουν 2 θέσεις.

Στη γραμμή που ακολουθεί εκτυπώστε ένα χαρακτήρα tab, το μήνυμα **Income:Y** όπου Y τα συνολικά έσοδα σε ευρώ από τη στάθμευση μοτοσυκλετών, και χαρακτήρα αλλαγής γραμμής. Το Y πρέπει να έχει χώρο για 4 ακέραια ψηφία και 2 δεκαδικά.

Εκτυπώστε παρόμοια μηνύματα για τα αυτοκίνητα, φορτηγά και λεωφορεία στις αντίστοιχες κατηγορίες.

Μετά την εκτύπωση του εισοδήματος για τα λεωφορεία, εκτυπώστε το μήνυμα **Total income:Y** όπου Y το σύνολο των εσόδων από όλα τα οχήματα (ίδιο φορμάρισμα με πριν), και χαρακτήρα αλλαγής γραμμής.

Ακολουθώντας, εκτυπώστε ένα χαρακτήρα αλλαγής γραμμής, το μήνυμα **Z% is moto income** και χαρακτήρα αλλαγής γραμμής. Το Z είναι το ποσοστό εσόδων από τη στάθμευση μοτοσυκλετών επί του συνόλου των εσόδων και πρέπει να έχει ακριβώς δύο δεκαδικά ψηφία.

Εκτυπώστε παρόμοια μηνύματα για τα αυτοκίνητα, φορτηγά και λεωφορεία στις επόμενες τρεις γραμμές, αντικαθιστώντας το **moto** με **car**, **truck** και **bus** αντίστοιχα.

Ολοκληρώστε αυτό το στάδιο και βεβαιωθείτε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Στάδιο 3: Διαχείριση πληρωμής (***)

Μετά τον υπολογισμό του ποσού πληρωμής για ένα όχημα, το πρόγραμμα πρέπει να ζητά το ποσό από τον πελάτη, να δέχεται την πληρωμή του και να υπολογίζει τα ρέστα που θα του δώσει.

Για την ακρίβεια, το πρόγραμμα εκτυπώνει το μήνυμα **Please pay X euros:** ακολουθούμενο από ένα κενό. Το X είναι το ποσό που πρέπει να πληρώσει ο πελάτης σε ευρώ κι εμφανίζεται με δύο δεκαδικά ψηφία.

Αφού διαβαστεί το ποσό που δίνει ο πελάτης, το πρόγραμμα πρέπει να επιβεβαιώσει ότι είναι αρκετό. Αν όχι, τότε το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα **Please pay another X euros:** ακολουθούμενο από ένα κενό. Το X είναι το υπόλοιπο που πρέπει να δώσει ο πελάτης ώστε να καλυφθεί το ποσό που χρωστάει. Η διαδικασία επαναλαμβάνεται μέχρι να καλυφθεί το ποσό.

Εφόσον ο πελάτης δώσει αρκετά χρήματα, το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα **You paid Y** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής. Το Y είναι το συνολικό ποσό που έδωσε τελικά ο πελάτης, κι έχει 2 δεκαδικά ψηφία.

Ακολουθώντας το πρόγραμμα υπολογίζει τα ρέστα που πρέπει να δώσει. Για ευκολία, υποθέτουμε ότι δίνει μόνο ευρώ, εικοσάλεπτα, πεντάλεπτα και λεπτά. Για τα ρέστα θα πρέπει να δίνεται ο μικρότερος δυνατός αριθμός κερμάτων. Με άλλα λόγια, κατά τον υπολογισμό των ρέστων δίνουμε όσο το δυνατό περισσότερα κέρματα μεγαλύτερης αξίας πριν αρχίσουμε να υπολογίζουμε τον αριθμό των κερμάτων αμέσως μικρότερης αξίας που θα δοθούν κ.ο.κ. Το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής και το μήνυμα **Please take your change:** ακολουθούμενο από ένα κενό.

Μετά, για κάθε υποδιαίρεση εκτυπώνει την ποσότητα, ένα κενό, την υποδιαίρεση και ένα κενό. Στο τέλος εκτυπώνει ένα χαρακτήρα αλλαγής γραμμής. Χρησιμοποιήστε **euros, twenties, fives** και **cents** για τις υποδιαίρεσεις. Αν κάποια ποσότητα είναι μηδέν, δεν εκτυπώνεται το αντίστοιχο μήνυμα.

Προσοχή: Στην περίπτωση που κάποιος μπει και βγει από το χώρο εντός του ίδιου λεπτού, τότε το ποσό πληρωμής είναι μηδενικό, και το πρόγραμμα εκτυπώνει απλά το μήνυμα **No payment required.** ακολουθούμενο από χαρακτήρα αλλαγής γραμμής. Σε αυτή την περίπτωση δεν εκτυπώνεται τίποτα από τα παραπάνω.

Σημειώσεις για την ακρίβεια υπολογισμών:

Όταν υπολογίζετε το ποσό που πρέπει να πληρώσει ένας οδηγός, πρέπει να το στρογγυλεύετε στα δύο δεκαδικά ψηφία. Για παράδειγμα, αν μια μοτοσυκλέτα παρκάρει για 10 λεπτά, τότε πρέπει να πληρώσει 0.13 κι όχι 0.125 ευρώ.

Προβλήματα ακρίβειας μπορεί να σας εμφανιστούν και κατά τον υπολογισμό των ρέστων. Για παράδειγμα, μπορεί το πρόγραμμα να υπολογίσει ότι τα ρέστα είναι 4.999999 ευρώ αντί για 5. Για να λύσετε το πρόβλημα, μπορείτε να κάνετε στρογγυλοποιήσεις χρησιμοποιώντας τις συναρτήσεις `round` ή `lround` ή κάποιο άλλο έξυπνο τρόπο.

Αν χρησιμοποιήσετε τις συναρτήσεις, γράψτε `y = round(x)` για να αναθέσετε στη μεταβλητή y την τιμή της μεταβλητής x στρογγυλοποιημένη (εννοείται εσείς δε θα χρησιμοποιήσετε τέτοια ονόματα μεταβλητών). Θα πρέπει επίσης να κάνετε `#include <math.h>` και όταν γράψετε την εντολή `gcc` να προσθέσετε στο τέλος και την επιλογή `-lm` (παύλα ελ εμ)

Στάδιο 4: Έλεγχος ορθότητας

Το πρόγραμμά σας πρέπει να λειτουργεί σωστά και να εκτυπώνει όλα τα μηνύματα και αποτελέσματα με τον τρόπο που σας περιγράφουμε. Για να μπορέσετε να ελέγξετε την ορθότητα θα σας δώσουμε ενδεικτικά αρχεία εισόδου και εξόδου. Μπορείτε να χρησιμοποιήσετε ανακατεύθυνση για να παράγετε τη δική σας έξοδο με βάση την είσοδο που σας δίνουμε:

Ας υποθέσουμε ότι το εκτελέσιμο πρόγραμμά σας λέγεται `hw1`, το ενδεικτικό αρχείο εισόδου λέγεται `input1.txt` και το αντίστοιχο αρχείο εξόδου που σας έχουμε δώσει λέγεται `stdout1.txt`

Η εντολή:

```
./hw1 < input1.txt > myout1.txt
```

εκτελεί το πρόγραμμά σας με την ενδεικτική είσοδο `input1.txt` και αποθηκεύει τα αποτελέσματα στο αρχείο εξόδου `myout1.txt` (αντί για την οθόνη)

Η εντολή

```
diff -b myout1.txt stdout1.txt
```

συγκρίνει το δικό σας αρχείο εξόδου με το δικό μας. Αν υπάρχουν διαφορές, τις εμφανίζει (γραμμή-γραμμή). Αν δεν υπάρχουν διαφορές, δεν κάνει τίποτα.

Πρέπει το πρόγραμμα που θα μας παραδώσετε να παράγει έξοδο που δεν έχει καθόλου διαφορές από τη δική μας.

Πώς να παραδώσετε τη δουλειά σας

Πριν παραδώσετε το πρόγραμμά σας, προσθέστε σε σχόλια στην αρχή του αρχείου τα πλήρη ονόματα και ΑΜ των μελών της ομάδας. Παρακαλούμε να γράφετε τα σχόλια ΜΟΝΟ με λατινικούς χαρακτήρες.

Κατασκευάστε ένα φάκελο με όνομα `hw1_epwnumero1_AM1_epwnumero2_AM2` και αντιγράψτε μέσα σε αυτόν το `hw1.c`

Πηγαίνετε στο φάκελο μέσα στον οποίο βρίσκεται το `hw1_epwnumero1_AM1_epwnumero2_AM2` που κατασκευάσατε και γράψτε την παρακάτω εντολή:

```
tar czf hw1_epwnumero1_AM1_epwnumero2_AM2.tar.gz hw1_epwnumero1_AM1_epwnumero2_AM2
```

Στείλτε email:

- στη διεύθυνση ce120lab@gmail.com
- αντίγραφο (CC) στον άλλο μέλος της ομάδας σας
- θέμα (subject) **CE120 hw1**
- και συνημμένο αρχείο το `hw1_epwnumero1_AM1_epwnumero2_AM2.tar.gz`