**ΓΡΑΠΤΗ ΕΡΓΑΣΙΑ 1**

**Θέμα 1ο**

Σχόλια επί του κώδικα της άσκησης.

Γραμ. 5, η επιλογή τον num1, num2 είναι τύπου double για περισσότερα ψηφία είσοδο.

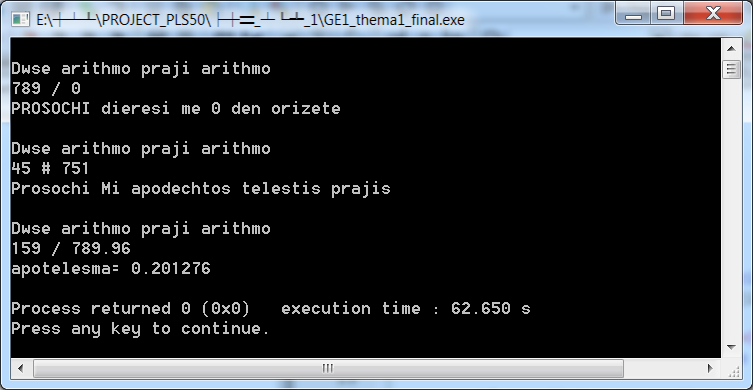
Γραμ. 6, η μεταβλητή praji είναι αναγκαστικά τύπου char.

Στις γραμμές 12-13 και 15-16 γίνεται ο έλεγχος λαθών. Η θέσης τους είναι τυχαία και θα μπορούσαν να αντιστραφούν χωρίς να δημιουργήσουν πρόβλημα στο κώδικα. Θεωρώ όμως ότι πρέπει να εκτελεστούν μετά την είσοδο των δεδομένων και πριν την εκτέλεση των πράξεων (για την τρέχουσα δομή του κώδικα).

Στις γραμμές 21-29 γίνεται έλεγχος για κάθε πράξη και η εκτύπωση του αποτελέσματος του οποίου η ακρίβεια συνδέεται άμεσα με την επιλογή του τύπου της μεταβλητής που έχουμε ορίσει στις γραμμές 22, 24, 26, 28.

Εναλλακτικά με τη χρήση του if θα μπορούσε να χρησιμοποιηθεί το if –else με το τελευταίο else να αποτελεί τον έλεγχο για τον τελεστή της πράξης.

Τέλος το do – while στις γραμμές 8-19 επιστρέφει στην είσοδο δεδομένων μετά από τον έλεγχο που γίνεται για εισαγωγή λανθασμένου τελεστή ή διαίρεση με μηδέν.

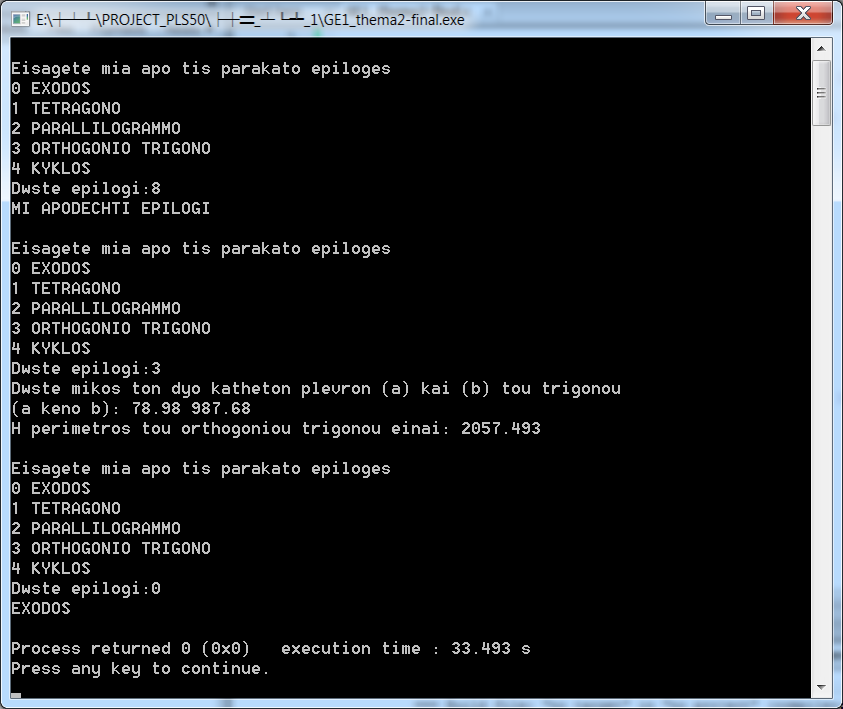


Στιγμιότυπο από την εκτέλεση του κώδικα της 1ης άσκησης.

**Θέμα 2ο**

Σχόλια επί του κώδικα της άσκησης.

Στο κώδικα που επισυνάπτεται παρατηρήθηκε το εξής φαινόμενο:   
Ι) όταν η choice (γραμ. 14) ορίζεται σαν char τότε   
    1) αν στο scanf (γραμ. 24) δηλωθεί %c ο κώδικας δεν τρέχει γιατί βγάζει όλες τις επιλογές "MI APODECHTI EPILOGI"   
    2) αν στο scanf (γραμ. 24) δηλωθεί %d ο κώδικας τρέχει όταν εισάγεις στο μενού χαρακτήρες κειμένου επιστρέφει "EXODOS" και όχι "MI APODECHTI EPILOGI" που είναι και το σωστό   
ΙΙ)  όταν η choice (γραμ. 14) ορίζεται σαν int τότε αν στο scanf (γραμ. 24) δηλωθεί %d ο κώδικας τρέχει και επιστρέφει τα σωστά μηνύματα ακόμα και όταν εισάγεις στο μενού χαρακτήρες κειμένου.   
Θεωρώ ότι η σωστότερη λύση με βάση τη δομή του συγκεκριμένου κώδικα είναι η ΙΙ (int και %d) εφόσον οι επιλογές που εισάγουμε είναι ακέραιοι αριθμοί.



Στιγμιότυπο από την εκτέλεση του κώδικα της 2ης άσκησης.

Η σταθερά pi θα μπορούσε να δηλωθεί είτε με define στην αρχή του κώδικα είτε με const στη συνάρτηση circle όπου και χρησιμοποιείται (γραμ. 81).

Εναλλακτικά θα μπορούσαν να χρησιμοποιηθούν τα μοντέλα if-else ή switch –case για την εκτέλεση του κυρίως κώδικα.

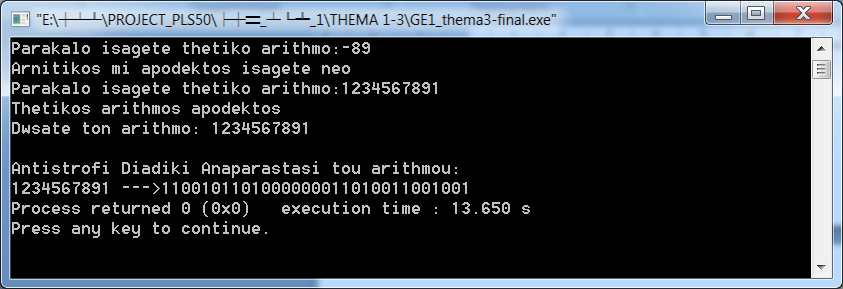
Όλα τα αποτελέσματα εκτυπώνονται με ακρίβεια τριών (3) δεκαδικών ψηφίων.

**Θέμα 3ο**

Σχόλια επί του κώδικα της άσκησης.

Στις γραμμές 5,6 γίνεται η δήλωση των μεταβλητών του κώδικα. Δηλώνοντας τον num τύπου int ο μέγιστός αριθμός που μπορούμε να εισάγουμε είναι o ακέραιος 2.147.483.647 που αντιστοιχεί στον μέγιστο 32bit αριθμό.

Ό κώδικας λειτουργεί είτε όταν στη γραμμή 20 η εντολή while (συνθήκη τερματισμού) δηλωθεί μεγαλύτερη του μηδενός (>0) είτε όταν οριστεί διάφορη του μηδενός (!=0).



Στιγμιότυπο από την εκτέλεση του κώδικα της 3ης άσκησης.

Ο κυρίως κώδικας εκτελείται στις γραμμές 20 έως 25. Το υπόλοιπο της διαίρεσης θα είναι πάντα 1 ή 0 ανάλογα αν ο αριθμός είναι μονός ή ζυγός. Το πρώτο υπόλοιπο που υπολογίζεται και εκτυπώνεται (γρ. 24) αντιστοιχεί στο λιγότερο σημαντικό ψηφίο (δεξιότερο, LSB) και το τελευταίο που υπολογίζεται και εκτυπώνεται (για τιμή αριθμού <0), αντιστοιχεί στο περισσότερο σημαντικό ψηφίο (αριστερότερο, MSB). Η σειρά εκτύπωσης λοιπόν είναι LSB🡪MSB που αντιστοιχεί στην αντίστροφή δυαδική αναπαράσταση του αριθμού.

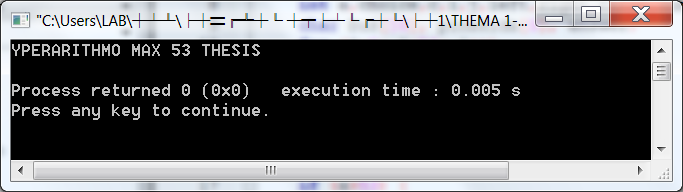
**Θέμα 4ο**

Σχόλια επί του κώδικα της άσκησης.

Γρ. 10-11 αρχικοποίηση του πίνακα με μηδενικές τιμές.

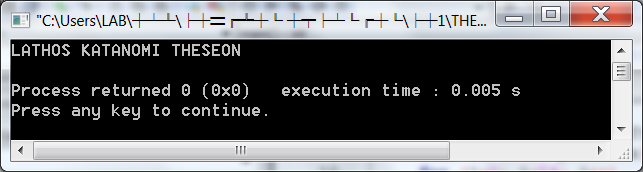
Γρ. 13-15 άνοιγμα αρχείου και διάβασμα δεδομένων: αριθμού κυκλοφορίας και αριθμού θέσεων λεωφορείου.

Γρ. 17-20 έλεγχος για αριθμό θέσεων μεγαλύτερο από 53 υπεράριθμου (1ο Στιγμιότυπο).



**1o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Έλεγχος για θέσεις μεγαλύτερες από 53.

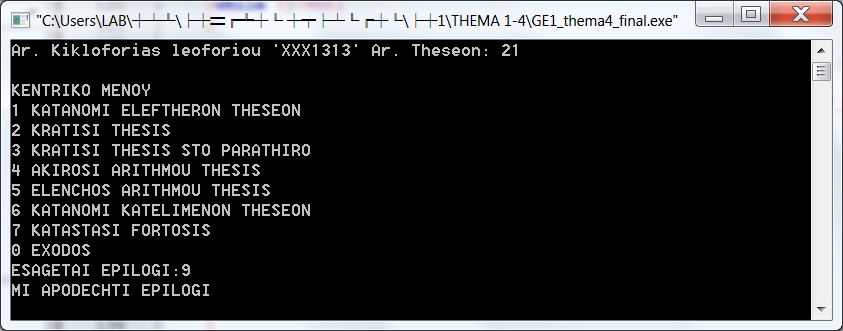
Γρ. 21-24 έλεγχος για συμμόρφωση της κατανομής θέσεων με τη μορφή 4xN+5. (2ο Στιγμιότυπο)



**2o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Έλεγχος για κατανομή θέσεων 4xN+5.

Γρ. 25-27 μετά τους παραπάνω ελέγχους τυπώνει τον αριθμό πινακίδας και τον αριθμό θέσης και εισέρχεται στο κύριο μενού του προγράμματος. (3o Στιγμιότυπο)

Γρ. 30-46 εμφάνιση του κεντρικού μενού του προγράμματος, αναμονή για εισαγωγή επιλογής από το χρήστη και έλεγχος συμβατότητας της επιλογής του.



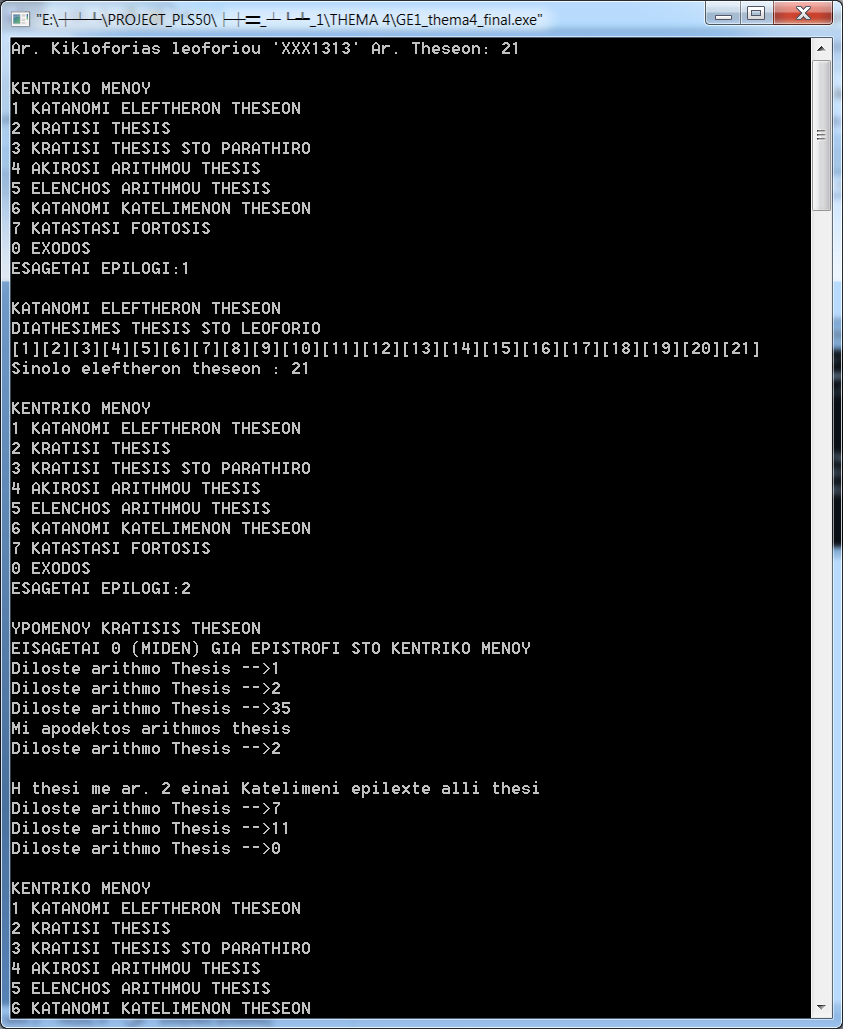
**Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Έλεγχος για μη προβλεπόμενη επιλογή χρήστη στο μενού.

**Ερώτημα 1.** Υλοποιείται στις γρ. 50-60. Εκτύπωση αποτελεσμάτων γρ. 53 και 59. (3o Στιγμιότυπο).

**Ερώτημα 2.** Υλοποιείται στις γρ. 62-79.

Έλεγχος υπεράριθμου γρ. 70-71. Έλεγχος καταλυμένης θέσης γρ. 73.

Γρ. 65-78, do – while: για επιστροφή στην επιλογή κράτησης νέας θέσης. Έξοδος στο κεντρικό μενού με μηδέν. (3o Στιγμιότυπο).

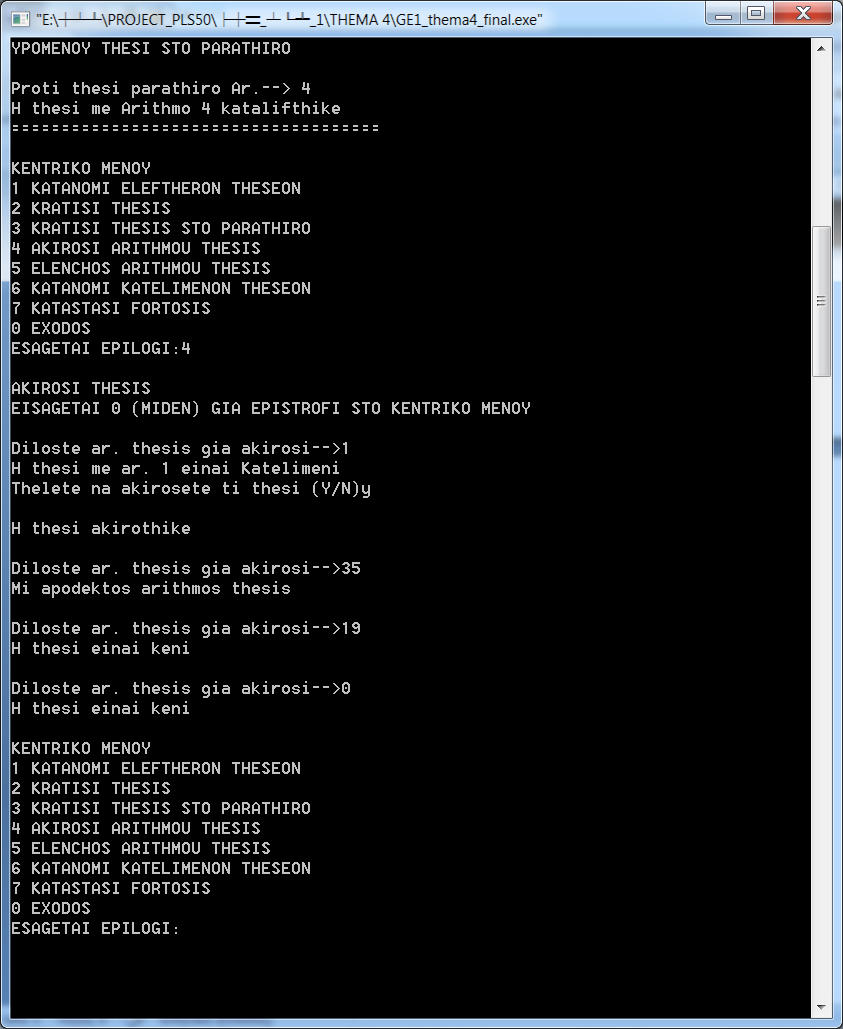


**3o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Ερώτημα 1 και 2

**Ερώτημα 3.** Υλοποιείται στις γρ. 81-97. Θέσεις χωρισμένες αριστερά (left-μονός αριθμός) και δεξιά (right-ζυγός αριθμός). Εύρεση της μικρότερης από τις δύο και κατάληψη της γρ. 93-95. (4o Στιγμιότυπο).

**Ερώτημα 4.** Υλοποιείται στις γρ. 99-121. Έλεγχος υπεράριθμου γρ. 107-108. Έλεγχος καταλυμένης θέσης και επιβεβαίωση ή μη ακύρωσης θέσης γρ. 111-117 (do – while).

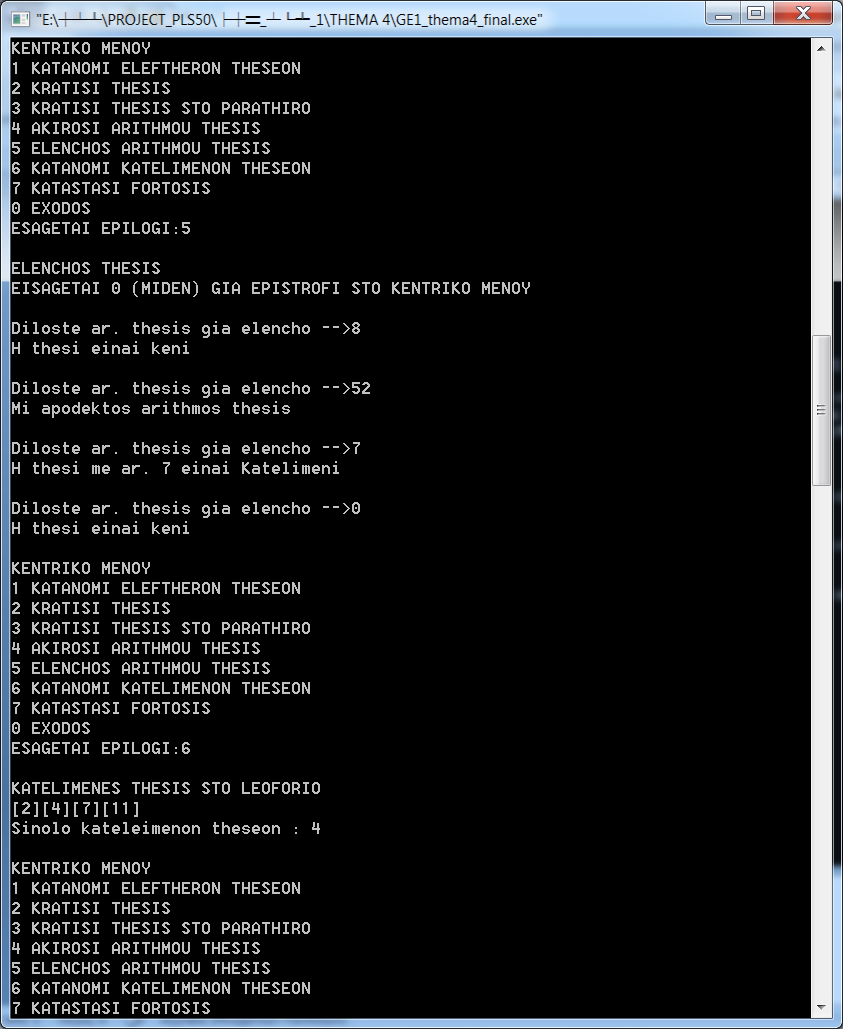
Έλεγχος κενής θέσης γρ. 118-119. Γρ. 102-120, do – while: για επιστροφή και συνέχιση ακύρωσης νέας θέσης. Έξοδος στο κεντρικό μενού με μηδέν. (4o Στιγμιότυπο).



**4o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Ερώτημα 3 και 4

**Ερώτημα 5.** Υλοποιείται στις γρ. 123-140. Έλεγχος υπεράριθμου γρ. 131-132. Έλεγχος καταλυμένης θέσης ή μη γρ. 134-138. Γρ. 126-139, do – while: για επιστροφή και συνέχιση ελέγχου νέας θέσης. Έξοδος στο κεντρικό μενού με μηδέν. (5o Στιγμιότυπο).

**Ερώτημα 6.** Υλοποιείται στις γρ. 142-151. Εκτύπωση αποτελεσμάτων γρ. 147 και 150. (5o Στιγμιότυπο).

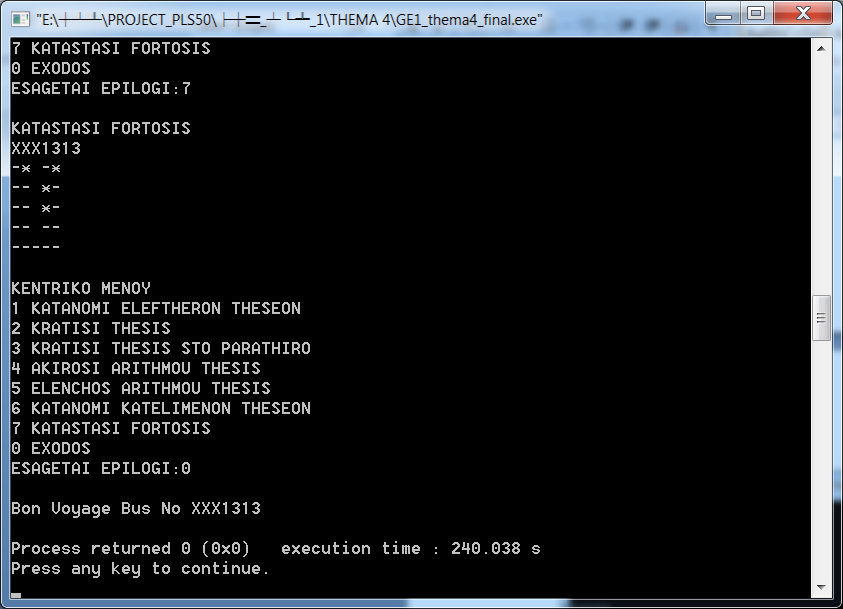


**5o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Ερώτημα 5 και 6

**Ερώτημα 7.** Υλοποιείται στις γρ. 153-168. Γρ. 156-158 μετατροπή των 0 και 1 σε – και \* αντίστοιχα για την εκτύπωση του σχεδιαγράμματος των θέσεων του λεωφορείου.

Γρ. 165-167 μετατροπή των - και \* σε 0 και 1 αντίστοιχα για την δυνατότητα περαιτέρω επεξεργασίας των θέσεων του λεωφορείου με βάση τις επιλογές του υπόλοιπου μενού (6o Στιγμιότυπο).

**Ερώτημα 8.** Υλοποιείται στις γρ. 47-49 και αποτελεί την έξοδο από το πρόγραμμα με ταυτόχρονή εκκίνηση της διαδικασίας εγγραφής στο αρχείο layout.txt της τελικής κατάστασης φόρτωσης του λεωφορείου (6o Στιγμιότυπο).



**6o Στιγμιότυπο** από την εκτέλεση του κώδικα της 4ης άσκησης. Ερώτημα 7 και 8

Δημιουργία και εγγραφή αρχείου layout.txt υλοποιείται στις γρ. 173-184. Γρ. 174-176 μετατροπή των 0 και 1 σε – και \* αντίστοιχα για την εκτύπωση του σχεδιαγράμματος της τελικής κατάστασης των θέσεων του λεωφορείου. Η εγγραφή στο αρχείο ενεργοποιείται με την έξοδο από το πρόγραμμα.

Γρ. 29-169 do – while: περιέχει το κεντρικό βρόγχο ελέγχου του κώδικα που δίνει τη δυνατότητα λειτουργίας του κεντρικού μενού του προγράμματος.