

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ Ι, ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2014-2015

ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ

Ποιος πρέπει να ολοκληρώσει αυτή την εργασία?

Φοιτητές έτους ≥ 2 που δεν ολοκλήρωσαν επιτυχώς την εργασία του ακαδημαϊκού έτους 2013-2014

Η εργασία μπορεί να γίνει είτε ατομικά είτε σε ομάδα των δύο ατόμων.

Προθεσμία

Τετάρτη, 7 Ιανουαρίου 2015, 23:59

Διαδικαστικά

Η προθεσμία παράδοσης είναι τελική - εκπρόθεσμες εργασίες δε γίνονται δεκτές.

Για να παραδώσετε την εργασία σας:

Δημιουργήστε ένα φάκελο με όνομα (με λατινικούς χαρακτήρες)

επώνυμο_όνομα_AEM_email

ή

επώνυμο1_όνομα1_AEM1_email1_επώνυμο2_όνομα2_AEM2_email2

αν η εργασία έχει γίνει σε ομάδα.

Μέσα σε αυτό το φάκελο **αντιγράψτε το αρχείο C που γράψατε.**

Δημιουργήστε ένα **αρχείο ZIP ή TGZ** το οποίο περιέχει το φάκελλο που κατασκευάσατε.

Η παράδοση γίνεται με email στο ce120lab@gmail.com. Μόνο το ένα μέλος της ομάδας πρέπει να στείλει την άσκηση. Το email πρέπει να συντάσσεται ως εξής:

Θέμα (Subject): CE120 project

Πρέπει να γράψετε το θέμα ακριβώς με αυτό τον τρόπο για να περάσει από τα κατάλληλα φίλτρα.

CC: το email του δεύτερου μέλους της ομάδας

Περιεχόμενο (Text): Τα ονόματα και ΑΕΜ των μελών της ομάδας

Επισυνάψεις (attachments): Το ZIP ή TAR ή TGZ αρχείο που δημιουργήσατε

ΜΗ ΣΥΜΠΕΡΙΛΑΒΕΤΕ ΤΟ ΛΕΞΙΚΟ ΣΤΟ ZIP/TAR/TGZ ΑΡΧΕΙΟ!

Περιγραφή παιχνιδιού

Θα υλοποιήσετε ένα παιχνίδι όπου ο υπολογιστής διαλέγει δύο λέξεις (ίδιου μήκους) από το λεξικό, κι ο παίκτης προσπαθεί να βρει ένα μονοπάτι από τη μία στην άλλη έτσι ώστε σε κάθε βήμα να αλλάζει ακριβώς ένα γράμμα, η νέα λέξη κάθε βήματος να ανήκει και αυτή στο λεξικό, και να έχει βρεθεί η λέξη-προορισμός σε περιορισμένο αριθμό βημάτων.

Στην αρχή του παιχνιδιού ο παίκτης προσδιορίζει πόσα γράμματα επιθυμεί να έχουν οι λέξεις που θα διαλέξει ο υπολογιστής, σε πόσα βήματα θα πρέπει να έχει βρεθεί το μονοπάτι καθώς και το όνομα του αρχείου που περιέχει το λεξικό.

Αφού το πρόγραμμα επιλέξει τυχαία τις δύο λέξεις του επιθυμητού μεγέθους (αν υπάρχουν), εμφανίζει στον παίκτη όλες τις λέξεις του λεξικού που απέχουν κατά ένα γράμμα από τη λέξη αφετηρία. Ο παίκτης διαλέγει μια από αυτές. Στο επόμενο βήμα, το πρόγραμμα εμφανίζει στον παίκτη όλες τις λέξεις του λεξικού που απέχουν κατά ένα γράμμα από την τελευταία λέξη που επιλέχθηκε. Στις επιλογές μπορεί να περιλαμβάνονται και λέξεις που έχουν ήδη επιλεγεί, και είναι ήδη στο μονοπάτι.

Αν βρεθεί η λέξη προορισμός πριν εξαντληθούν τα επιτρεπτά βήματα, εκτυπώνεται μήνυμα επιτυχίας

Αν εξαντληθούν τα βήματα πριν βρεθεί η λέξη προορισμός, το πρόγραμμα εκτυπώνει μήνυμα αποτυχίας. Προαιρετικά, μπορείτε να προσθέσετε την παρακάτω λειτουργικότητα σε αυτό το σημείο: αν ο παίκτης δεν έχει βρει τη λέξη προορισμό, τότε το πρόγραμμα ψάχνει μόνο του αναδρομικά (με χρήση backtracking) να δει αν υπάρχει λύση, κι αν ναι, την εμφανίζει στην οθόνη. Σημείωση: Αν προσθέσετε αυτή την επιπλέον λειτουργικότητα θα χρειαστεί επιπλέον πεδίο στο struct για τις λίστες σας: Για να λειτουργήσει σωστά η αναδρομή πρέπει να μαρκάρονται οι λέξεις που έχουν ήδη ελεγχθεί για να μην ξαναμπούν στο μονοπάτι και δημιουργηθεί "κύκλος". Αρκεί μια boolean μεταβλητή για αυτό.

Σε κάθε περίπτωση, το πρόγραμμα ρωτά τον παίκτη αν θέλει να ξαναπαίξει. Αν ναι, διαλέγει δύο νέες λέξεις κι επαναλαμβάνει το παιχνίδι (με ίδιο ή διαφορετικό μήκος λέξεων / πλήθος βημάτων, ανάλογα με την προτίμησή σας), διαφορετικά τερματίζει.

Υλοποίηση

Εκκίνηση παιχνιδιού

Για την εκκίνηση του παιχνιδιού χρειαζόμαστε τρεις πληροφορίες: το μήκος των δύο λέξεων, σε πόσα βήματα πρέπει να έχουμε πάει από τη λέξη-αφετηρία στη λέξη-προορισμό, και το όνομα του αρχείου-λεξικού. Το αρχείο-λεξικό περιέχει μια μεγάλη συλλογή λέξεων.

Και οι τρεις πληροφορίες δίνονται από τη γραμμή εντολής ως ορίσματα του προγράμματος, με τη σειρά που περιγράφεται στην προηγούμενη παράγραφο.

Γράψτε μια συνάρτηση σκοπός της οποίας είναι να επεξεργαστεί τα ορίσματα του προγράμματος, να επιβεβαιώσει ότι έχουν δοθεί έγκυρες τιμές και να επιστρέφει (μέσω της λίστας παραμέτρων της συνάρτησης) το μέγεθος της λέξης και τον αριθμό των γύρων. Η συνάρτηση επιστρέφει ένα κωδικό λάθους ανάλογα με το αν η επεξεργασία των ορισμάτων ήταν επιτυχής ή όχι.

Για την ακρίβεια, η συνάρτηση αυτή :

- Ελέγχει αν έχει δοθεί ο σωστός αριθμός ορισμάτων προγράμματος. Αν έχουν δοθεί λιγότερα από όσα πρέπει, εκτυπώνει το μήνυμα **"Insufficient arguments."**, ένα χαρακτήρα αλλαγής γραμμής, οδηγίες εκτέλεσης του προγράμματος (λεπτομέρειες πιο κάτω) κι επιστρέφει 0. Αν έχουν δοθεί περισσότερα από όσα πρέπει, εκτυπώνει το μήνυμα **"Too many arguments."**, ένα χαρακτήρα αλλαγής γραμμής, οδηγίες εκτέλεσης του προγράμματος κι επιστρέφει 0.

- Επιβεβαιώνει ότι δόθηκε θετικός αριθμός βημάτων, κι αν όχι εκτυπώνει το μήνυμα **"You may not have**

fewer than one steps. Specify a different number of steps: " και διαβάζει από το πληκτρολόγιο νέα τιμή.

- Αφού ανατεθούν οι τιμές για το μήκος και τα βήματα σε κατάλληλες μεταβλητές, επιστρέφει 1.

Οι οδηγίες εκτέλεσης του προγράμματος, οι οποίες εκτυπώνονται σε περίπτωση που έχει δοθεί λάθος αριθμός ορισμάτων είναι:

```
<exec> LENGTH STEPS DICTIONARY
    LENGTH: requested length of word.
    STEPS: requested number of steps. Must be >0
    DICTIONARY: name of dictionary file
```

όπου **<exec>** το όνομα του εκτελέσιμου προγράμματός σας.

Οι τρεις τελευταίες γραμμές ξεκινούν ένα tab πιο δεξιά από την πρώτη, και μετά την τελευταία γραμμή ακολουθεί χαρακτήρας αλλαγής γραμμής.

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

Δομές δεδομένων

Θα χρειαστείτε τρεις βασικές δομές δεδομένων για το πρόγραμμα:

- **Μια διασυνδεδεμένη λίστα** στην οποία αποθηκεύονται οι λέξεις του μονοπατιού κάθε χρονική στιγμή.
 - * Οι λέξεις τοποθετούνται στη λίστα με τη σειρά που επιλέγονται από τον παίκτη, και κάθε νέα λέξη προστίθεται στο τέλος της λίστας
- **Μια διασυνδεδεμένη λίστα** στην οποία αποθηκεύονται όλες οι λέξεις του συγκεκριμένου μήκους.
 - * Δε μας ενδιαφέρει η σειρά σε αυτή τη λίστα.
- **Μια διασυνδεδεμένη λίστα** στην οποία αποθηκεύονται όλες οι λέξεις που είναι διαθέσιμες προς επιλογή για το επόμενο βήμα.
 - * Οι λέξεις τοποθετούνται στη λίστα ταξινομημένες.

Θα χρησιμοποιήσετε το ίδιο struct και για τις τρεις λίστες. Και οι τρεις είναι διπλά διασυνδεδεμένες, κυκλικές, με τερματικό κόμβο (sentinel).

Σε κάθε κόμβο της λίστας αποθηκεύονται:

- Ο τύπος της λίστας (αν είναι ταξινομημένη ή όχι - χρησιμοποιήστε τύπο enum)
- Τα δεδομένα του κόμβου:
 - * Αν πρόκειται για τον τερματικό κόμβο, σε αυτό το πεδίο αποθηκεύετε το πλήθος λέξεων
 - * Διαφορετικά, σε αυτό το πεδίο αποθηκεύετε τη λέξη ως τύπο δείκτη προς χαρακτήρα.
 - * hint: Χρησιμοποιήστε τύπο union
- Δείκτες προς τον προηγούμενο κι επόμενο κόμβο της λίστας.

Γράψτε μια συνάρτηση η οποία παίρνει ως παραμέτρους το επιθυμητό μήκος λέξης και το όνομα του αρχείου-λεξικού, κατασκευάζει μία λίστα με όλες τις λέξεις αυτού του μήκους (το αρχείο-λεξικό μπορεί περιέχει και λέξεις διαφορετικών μηκών) κι επιστρέφει δείκτη προς την κεφαλή της. Σημειώστε πως υπάρχει πιθανότητα η λίστα να μην περιέχει αρκετές λέξεις ώστε να μπορεί να παιχτεί το παιχνίδι (δηλαδή περιέχει λιγότερες από 2). Σε αυτή την περίπτωση, η συνάρτησή σας πρέπει να επιστρέφει NULL.

Σας συνιστούμε ανεπιφύλακτα να γράψετε συναρτήσεις για τις λειτουργίες λιστών (πχ εισαγωγή, εκτύπωση) και όχι να χρησιμοποιήσετε inline κώδικα. Οι συναρτήσεις σας θα πρέπει να μπορούν να λειτουργήσουν σωστά ανάλογα με τον τύπο της λίστας. Για παράδειγμα, στη συνάρτηση εισαγωγής λέξης σε λίστα, θα πρέπει να ελέγχετε τον τύπο της λίστας, κι αν αυτή είναι ταξινομημένη να εισάγετε τη λέξη στη σωστή θέση λεξικογραφικά, ενώ αν δεν είναι να εισάγετε τη λέξη στο κατάλληλο σημείο (αρχή ή τέλος της λίστας). Οι μόνες συναρτήσεις που επιβάλλεται να ελέγχουν τον τύπο είναι η συνάρτηση εισαγωγής λέξεων και η συνάρτηση εκτύπωσης. Οι υπόλοιπες (πχ συνάρτηση αφαίρεσης λέξης) δε χρειάζεται απαραίτητα να κάνουν τέτοιο έλεγχο.

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

Βασική λειτουργία παιχνιδιού

Στο πρώτο βήμα, το παιχνίδι εκτυπώνει τα αρχικά στοιχεία:

Starting word: X

Target word: Y

Try to go from "X" to "Y" in no more than Z steps

όπου πριν τις λέξεις Starting και Target υπάρχει χαρακτήρας tab, X είναι η λέξη-αφετηρία, Y είναι η λέξη-προορισμός, Z είναι το πλήθος βημάτων.

Σε κάθε επόμενο βήμα, το παιχνίδι προχωρά ως εξής:

Εκτυπώνει το μήνυμα "Progress so far: L", ακολουθούμενο από χαρακτήρα αλλαγής γραμμής, όπου L είναι η λίστα με τις λέξεις που βρίσκονται μέχρι τώρα στο μονοπάτι. Η λίστα εκτυπώνεται ως εξής, αν υποθέσουμε ότι αποτελείται από τις λέξεις L1, L2, L3: L1 --> L2 --> L3

Εκτυπώνει το μήνυμα "Remaining steps: S", ακολουθούμενο από χαρακτήρα αλλαγής γραμμής, όπου S είναι το πλήθος βημάτων που απομένουν.

Εκτυπώνει το μήνυμα "Target: Y", ακολουθούμενο από χαρακτήρα αλλαγής γραμμής, όπου Y είναι η λέξη-προορισμός (για να μην την ξεχάσει ο παίκτης)

Εκτυπώνει χαρακτήρα αλλαγής γραμμής.

Εκτυπώνει το μήνυμα "Choose one of the following words: ", ακολουθούμενο από χαρακτήρα αλλαγής γραμμής

Εκτυπώνει τα περιεχόμενα της λίστας με τις λέξεις που απέχουν κατά ένα γράμμα από την τρέχουσα λέξη (τελευταία επιλεγμένη). Η λίστα εκτυπώνεται με τις λέξεις τις να εμφανίζονται σε 4 στήλες με ένα κενό χαρακτήρα να διαχωρίζει διπλανές στήλες. Υπάρχει περίπτωση η λίστα να μην περιέχει λέξεις. Σε αυτή την περίπτωση, το πρόγραμμα εκτυπώνει "There are no words to choose from. Sorry!" και τελειώνει αυτός ο γύρος.

Εκτυπώνει χαρακτήρα αλλαγής γραμμής.

Εκτυπώνει το μήνυμα "Enter selection: "

Διαβάζει από το πληκτρολόγιο μία λέξη. Αν αυτή δεν είναι μία από τις διαθέσιμες λέξεις προς επιλογή, τότε το πρόγραμμα εκτυπώνει το μήνυμα "ERROR: Your selection is not available. Try again" και επαναλαμβάνονται όλα τα παραπάνω βήματα (από το Progress so far και μετά). Το πλήθος βημάτων που απομένουν δεν πρέπει να μειωθεί σε αυτή την περίπτωση. Αν η λέξη που διαβάστηκε είναι μια από τις διαθέσιμες, τότε προστίθεται στη λίστα-μονοπάτι και το παιχνίδι συνεχίζει με τον ίδιο τρόπο (από το Progress so far και μετά).

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

Έλεγχος νικητή

Κάθε φορά που επιλέγεται μια νέα λέξη, πρέπει να ελέγχετε αν πρόκειται για τη λέξη-στόχο κι αν έχουν τελειώσει ή όχι τα διαθέσιμα βήματα.

Αν έχει κερδίσει ο παίκτης (βρέθηκε η λέξη-στόχος), **το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής** και το μήνυμα **"GOOD WORD! L"** όπου L η λίστα-μονοπάτι, και χαρακτήρα αλλαγής γραμμής.

Αν έχει χάσει ο παίκτης (δεν υπάρχουν άλλα βήματα), **το πρόγραμμα εκτυπώνει χαρακτήρα αλλαγής γραμμής** το μήνυμα **"TOO BAD :(L"** όπου L η λίστα-μονοπάτι, ακολουθούμενη από χαρακτήρα αλλαγής γραμμής και μετά το μήνυμα **"Target was X"** όπου X η λέξη-στόχος, κι ένα χαρακτήρα αλλαγής γραμμής. Αν ολοκληρώσετε το προαιρετικό κομμάτι, τότε μετά το χαρακτήρα αλλαγής γραμμής, το πρόγραμμα πρέπει είτε να εκτυπώνει το μήνυμα **"A solution is S"** όπου S το μονοπάτι που βρέθηκε αναδρομικά, είτε να εκτυπώνει **"There is no solution"**. Και στις δύο περιπτώσεις ακολουθεί χαρακτήρας αλλαγής γραμμής.

Σε κάθε περίπτωση τερματισμού του παιχνιδιού δίνεται η ευκαιρία στον παίκτη να ξαναπαίξει. **Το πρόγραμμα εκτυπώνει** ένα χαρακτήρα αλλαγής γραμμής και το μήνυμα **"Play again? (y/n): "** και **διαβάζει από το πληκτρολόγιο** ένα χαρακτήρα.

Αν αυτός είναι οτιδήποτε εκτός y/Y ή n/N τότε το πρόγραμμα εκτυπώνει το μήνυμα **"Please enter y for yes or n for no."**, ένα χαρακτήρα αλλαγής γραμμής, το μήνυμα **"Play again? (y/n): "** και ξαναδιαβάζει την απάντηση του παίκτη. Το βήμα αυτό επαναλαμβάνεται μέχρις ότου να δοθεί έγκυρη απάντηση.

Αν η απάντηση είναι y ή Y τότε ξεκινά καινούργιο παιχνίδι. Το μήκος της λέξης και πλήθος βημάτων μπορούν να παραμείνουν ίδια, ή μπορείτε να ζητήσετε νέες τιμές από τον παίκτη. Αν η απάντηση είναι n ή N, το πρόγραμμα τερματίζει.

Γενικές απαιτήσεις:

- Απαγορεύεται αυστηρά η χρήση καθολικών μεταβλητών
- Χρησιμοποιήστε συναρτήσεις για καλά ορισμένες λειτουργίες του προγράμματος. Θα δοθεί ιδιαίτερη σημασία στο πώς έχετε οργανώσει τον κώδικά σας.
- Πρέπει να χρησιμοποιείτε συναρτήσεις από τη string.h οπουδήποτε κάνετε διαχείριση συμβολοσειρών.
- Φροντίστε να αποδεσμεύετε πάντα σωστά τη δυναμικά δεσμευμένη μνήμη.
- Ελέγχετε πάντα τα αποτελέσματα των συναρτήσεων. Για παράδειγμα, αν η συνάρτηση δημιουργίας της αρχικής λίστας-λέξεων επιστρέφει NULL επειδή δεν υπάρχουν αρκετές λέξεις του δεδομένου μήκους, θα πρέπει να κάνετε σχετικό έλεγχο στη main.
- Το πρόγραμμά σας πρέπει να έχει αποτελεσματικά σχόλια, καλά ονόματα μεταβλητών, σωστή στοίχιση και γενική μορφή κώδικα.
- Το πρόγραμμά σας πρέπει να κάνει compile χωρίς λάθη ή προειδοποιήσεις και να τρέχει σε linux.
- Οτιδήποτε εκτυπώνεται στην οθόνη πρέπει να είναι ακριβώς όπως περιγράφεται στην εκφώνηση.
- Κάθε επιλογή λέξεων αφετηρίας/προορισμού πρέπει να γίνεται τυχαία

Βοήθεια

Στο αρχείο `getWord.c` σας δίνουμε την υλοποίηση μιας συνάρτησης με όνομα `getWord` η οποία κάθε φορά που καλείται διαβάζει την επόμενη λέξη από ένα αρχείο (το όνομα του οποίου δίνεται ως παράμετρος στη συνάρτηση). Η `getWord` επιστρέφει τη λέξη που διάβασε, ως δείκτη σε δυναμικά δεσμευμένη συμβολοσειρά, μεγέθους `WORDLEN`. Επιστρέφει `NULL` αν δεν υπάρχουν άλλες λέξεις στο αρχείο. Χρησιμοποιήστε τη ως έχει, χωρίς αλλαγές. Μην ξεχάσετε αργότερα (όταν δε χρειάζονται πια) να κάνετε `free` τις λέξεις που έχετε πάρει από τη `getWord`. Επίσης, θα σας παρέχουμε δύο αρχεία με λέξεις, ένα μικρού μεγέθους για εύκολο debugging κι ένα πλήρες για εκτεταμένα τεστ.

Παραδείγματα

Δείτε μερικά screenshots από την εκτέλεση του παιχνιδιού:

```
[vana ~]% ./a.out 5 20 largedict.txt
```

```
Starting word: riots
```

```
Target word: dirks
```

```
Try to go from "riots" to "dirks" in no more than 20 steps
```

```
Progress so far:      riots
```

```
Remaining steps: 20
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
rifts roots ryots
```

```
Enter selection: rifts
```

```
Progress so far:      riots --> rifts
```

```
Remaining steps: 19
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
gifts lifts rafts riffs  
sifts
```

```
Enter selection: gifts
```

```
Progress so far:      riots --> rifts --> gifts
```

```
Remaining steps: 18
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
giltz girts gists lifts  
rifts sifts
```

```
Enter selection: girts
```

```
Progress so far:      riots --> rifts --> gifts --> girts
```

```
Remaining steps: 17
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
airts dirts gifts giltz  
girds girls girns giros  
girth gists
```

```
Enter selection: dirst
```

```
ERROR: Your selection is not available. Try again
```

```
Progress so far:      riots --> rifts --> gifts --> girts
```

```
Remaining steps: 17
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
airts dirts gifts giltz  
girds girls girns giros  
girth gists
```

```
Enter selection: dirts
```

```
Progress so far:      riots --> rifts --> gifts --> girts --> dirts
```

```
Remaining steps: 16
```

```
Target: dirks
```

```
Choose one of the following words:
```

```
airts darts diets dintz  
dirks dirls dirty girts
```

```
Enter selection: dirks
```

```
GOOD WORK! riots --> rifts --> gifts --> girts --> dirts --> dirks
```

```
Play again? (y/n): ☐
```

Παράδειγμα 1: Επιτυχημένο παιχνίδι - παρατηρήστε και το σημείο που έγινε επιλογή λέξης που δεν υπήρχε στις διαθέσιμες.

```
[vana ~]% ./a.out 4 3 largedict.txt
```

Starting word: kino

Target word: pony

Try to go from "kino" to "pony" in no more than 3 steps

Progress so far: kino

Remaining steps: 3

Target: pony

Choose one of the following words:

fino keno kilo kina
kind kine king kink
kins lino vino wino

Enter selection: lino

Progress so far: kino --> lino

Remaining steps: 2

Target: pony

Choose one of the following words:

fino leno lido limo
line ling link linn
lins lint liny vino
wino

Enter selection: liny

Progress so far: kino --> lino --> liny

Remaining steps: 1

Target: pony

Choose one of the following words:

lily limy line ling
link linn lino lins
lint luny piny tiny
viny winy

Enter selection: piny

TOO BAD :(kino --> lino --> liny --> piny

Target was: pony

Play again? (y/n):

Παράδειγμα 2: Αποτυχημένο παιχνίδι (αχ, να είχε ένα ακόμη βήμα...)

```
[vana ~]% ./a.out 5 20 largedict.txt
```

Starting word: schul

Target word: franc

Try to go from "schul" to "franc" in no more than 20 steps

Progress so far: schul

Remaining steps: 20

Target: franc

There are no words to choose from. Sorry!

Play again? (y/n):

Παράδειγμα 3: Κενή λίστα λέξεων με απόσταση 1 από την επιλεγμένη.

```
[vana ~]% ./a.out
```

Insufficient arguments

./a.out LENGTH STEPS DICTIONARY

LENGTH: requested length of word.

STEPS: requested number of steps. Must be >0

DICTIONARY: name of dictionary file

```
[vana ~]%
```

Παράδειγμα 4: Λάθος κλήση προγράμματος