

# HY-100: Εισαγωγή στην Επιστήμη Υπολογιστών

## 4η σειρά ασκήσεων

### 1 Συχνότητα

Γράψτε ένα πρόγραμμα (hw4a.c) που θα διαβάζει από το πληκτρολόγιο έναν αριθμό  $N$ , στη συνέχεια θα διαβάζει  $N$  λέξεις, θα υπολογίζει την πιο συχνή λέξη ανάμεσα στις  $N$  και θα την τυπώνει. Το πρόγραμμά σας πρέπει να δεσμεύει δυναμικά τη μνήμη που θα χρησιμοποιήσει για να αποθηκεύσει τις λέξεις και τη συχνότητά τους. Μπορείτε να υποθέσετε ότι κάθε λέξη έχει το πολύ 50 χαρακτήρες.

Δίνεται το παρακάτω παράδειγμα εκτέλεσης όπου φαίνεται η είσοδος από το πληκτρολόγιο με κόκκινο και η έξοδος του προγράμματος με μπλε χρώμα.

```
$ ./a.out
10
hello
world
goodbye
world
thanks
for
all
hello
the
fish
hello
$
```

Για να δοκιμάσετε το πρόγραμμά σας δίνονται δυο παραδείγματα εισόδου μαζί με την εκφώνηση της άσκησης. Στην περιοχή σας στο εργαστήριο εκτελέστε την παρακάτω εντολή για να κατεβάσετε και να αποθηκεύσετε τα αρχεία που περιέχουν την ενδεικτική είσοδο.

```
$ wget http://www.csd.uoc.gr/~hy100/hw4in1.txt
$ wget http://www.csd.uoc.gr/~hy100/hw4in2.txt
```

Μπορείτε μετά να χρησιμοποιήσετε το σύμβολο `<` για να δώσετε στο πρόγραμμά σας τα περιεχόμενα του αρχείου σαν να τα είχατε πληκτρολογήσει.

```
$ ./a.out < hw4in1.txt
hello
$ ./a.out < hw4in2.txt
of
$
```

### 2 Λαβύρινθος

Γράψτε ένα πρόγραμμα (hw4b.c) που δημιουργεί λαβυρίνθους με αναδρομή. Για να φτιάξετε ένα λαβύρινθο, ακολουθήστε τον παρακάτω αλγόριθμο:

1. Ξεκινήστε με ένα συμπαγές τετράγωνο  $N \times M$ .
2. Στη συνέχεια, τοποθετήστε την έξοδο του λαβυρίνθου στο σημείο (0,1), “αδειάζοντας” αυτό το σημείο και το σημείο (1,1).

3. Για τη θέση που βρίσκεστε -αρχικά (1,1)- υπολογίστε τις πιθανές κατευθύνσεις (Πάνω, Κάτω, Δεξιά, ή Αριστερά) προς τις οποίες μπορείτε να σκάψετε. Για να είναι δυνατή η κίνηση προς μια κατεύθυνση πρέπει:
  - Να μην “τρυπήσει” κάποιος από τους εξωτερικούς τοίχους.
  - Να υπάρχουν 2 γεμάτα τετράγωνα προς εκείνη την κατεύθυνση.
4. Διαλέξτε μια κατεύθυνση από τις παραπάνω στην τύχη και προχωρήστε “σκάβοντας” ώστε να αδειάσουν δύο τετράγωνα προς αυτή την κατεύθυνση.
5. Αν δεν μπορείτε να “σκάψετε” προς καμία κατεύθυνση, κάντε 2 βήματα πίσω και ξαναπροσπαθήστε.
6. Επαναλάβετε τα παραπάνω βήματα, μέχρι να επιστρέψετε στην είσοδο.

Για παράδειγμα, με  $N = 7$  και  $M = 9$ , θα είναι αρχικά όλα τα σημεία του τετραγώνου γεμάτα:

```

XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

```

Τοποθετούμε την είσοδο του λαβυρίνθου:

```

X XXXXXXX
X XXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

```

Διαλέγουμε μια κατεύθυνση στην τύχη π.χ., Κάτω:

```

X XXXXXXX
X XXXXXXX
X XXXXXXX
X XXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

```

Συνεχίζουμε έτσι, μέχρι να τελειώσει ο λαβύρινθος:

```

X XXXXXXX
X X      X
X XXXXX X
X  X    X
XXX X X X
X      X X
XXXXXXXXXX

```

Το πρόγραμμα θα πρέπει να διαβάζει το  $N$  και το  $M$  από το πληκτρολόγιο (χωρίς να τυπώνει κάποιο μήνυμα), να υπολογίζει τον λαβύρινθο αναδρομικά με τον παραπάνω αλγόριθμο και να τον τυπώνει στην οθόνη, χρησιμοποιώντας τους χαρακτήρες 'X' και ' ' όπως στο παράδειγμα παραπάνω. Το πρόγραμμα πρέπει να τυπώνει μόνο το τελικό αποτέλεσμα και όχι τις ενδιάμεσες καταστάσεις του λαβυρίνθου. Χρησιμοποιήστε τη συνάρτηση `malloc` για να δεσμεύσετε αρκετό χώρο για το λαβύρινθο, και τη συνάρτηση `free` για να αποδεσμεύσετε τη μνήμη όταν δεν χρειάζεται πλέον.

## Οδηγίες

Για τη μεταγλώττιση των προγραμμάτων που ζητούνται θα πρέπει να χρησιμοποιήσετε το `gcc` με τις παρακάτω παραμέτρους:

```
gcc -ansi -pedantic -Werror
```

Για ευκολία, δημιουργήθηκε στα μηχανήματα του εργαστηρίου το script `gcc100` που καλεί τον `gcc` με αυτές τις παραμέτρους.

Για να παραδώσετε τα αρχεία με τις λύσεις στα ερωτήματα της άσκησης, ακολουθήστε τις οδηγίες που βρίσκονται στη σελίδα <http://www.csd.uoc.gr/~hy100/submit-howto-gr.html>.