ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ (2005-06)

Άσκηση 4

Έστω ένας ορθογώνιος λαβύρινθος πλάτους <width> και ύψους <height>. Ο λαβύρινθος αυτός αποτελείται από <width>×<height> κελιά, στα οποία θα αναφερόμαστε, στη συνέχεια, με έναν αύξοντα αριθμό από το 1 έως το <width>×<height>. Το κελί με αριθμό 1, που είναι και η είσοδος του λαβυρίνθου, είναι το πρώτο αριστερά κελί της πρώτης γραμμής. Η αρίθμηση συνεχίζεται με τα κελιά δεξιότερα στην πρώτη γραμμή, με το τελευταίο να είναι το υπ΄ αριθμόν <width> κελί. Το πρώτο κελί της δεύτερης γραμμής είναι το <width>+1 και ούτω καθεξής μέχρι το δεξιότερο κελί της τελευταίας γραμμής που είναι το κελί υπ΄ αριθμόν <width>×<height> και το οποίο είναι η έξοδος του λαβυρίνθου. Οι δίοδοι μεταξύ γειτονικών κελιών, οριζόντια ή κατακόρυφα, είναι είτε ανοικτές, οπότε κάποιος μπορεί να περάσει από το ένα κελί στο άλλο, είτε κλειστές, οπότε η διάβαση δεν είναι δυνατή. Παράδειγμα:

```
          <----<width>=6---->
   ^      +  +--+--+--+--+--+
   |      | 1  2| 3  4| 5  6|
   |      +--+  +--+  +  +  +
   |      | 7| 8  9|10 11|12|
<height>=4 +  +  +  +--+--+  +
   |      |13|14|15|16 17|18|
   |      +  +  +  +  +  +  +
   |      |19 20|21 22|23 24|
   v      +--+--+--+--+--+  +
```

Γράψτε ένα πρόγραμμα C (έστω ότι το πηγαίο αρχείο του ονομάζεται "escape.c") το οποίο, για δεδομένο λαβύρινθο, να βρίσκει διαδρομή μεταξύ των κελιών που να μπορεί να οδηγήσει από την είσοδο του λαβυρίνθου στην έξοδό του. Για να έχετε διαθέσιμους όσους λαβύρινθους θέλετε, κατεβάστε το εκτελέσιμο πρόγραμμα (για τα Suns του Τμήματος) που θα βρείτε στο

http://www.di.uoa.gr/~ip/hwprogs/labyrinth

Το πρόγραμμα labyrinth υπάρχει διαθέσιμο σε εκτελέσιμη μορφή, στην ίδια θέση που βρίσκεται η έκδοση για τα Suns, και για άλλες πιθανές πλατφόρμες, για Windows (labyrinth-win.exe), για Linux (labyrinth-lin), και για MacOS-X (labyrinth-mac). Για ευκολία στο κατέβασμα, υπάρχουν, στην ίδια θέση, και οι συμπιεσμένες, σε zip format, εκδόσεις των εκτελέσιμων προγραμμάτων, για τα Suns (labyrinth.zip), για Windows (labyrinth-win.zip), για Linux (labyrinth-lin.zip) και για MacOS-X (labyrinth-mac.zip).

Το πρόγραμμα αυτό όταν καλείται με δύο ορίσματα <width> και <height> δημιουργεί ένα λαβύρινθο με τη διάσταση αυτή και εκτυπώνει στην έξοδο τα δεδομένα του λαβυρίνθου. Συγκεκριμένα στην πρώτη γραμμή εκτυπώνει το <width> και το <height>. Σε κάθε μία από τις επόμενες γραμμές εκτυπώνει ένα πλήθος από ζευγάρια αριθμών $\langle\text{cell}\rangle_1$ $\langle\text{cell}\rangle_2$, που καθένα από τα οποία δηλώνει ότι η δίοδος μεταξύ του κελιού με αύξοντα αριθμό $\langle\text{cell}\rangle_1$ και αυτού με αύξοντα αριθμό $\langle\text{cell}\rangle_2$ είναι ανοικτή. Δεν θα εμφανιστεί, στην περίπτωση αυτή και το ζευγάρι $\langle\text{cell}\rangle_2$ $\langle\text{cell}\rangle_1$ παρότι η μετάβαση είναι δυνατή είτε κινούμαστε από το $\langle\text{cell}\rangle_1$ στο $\langle\text{cell}\rangle_2$ είτε από το $\langle\text{cell}\rangle_2$ $\langle\text{cell}\rangle_1$.

Ένα παράδειγμα εκτέλεσης είναι το εξής:

```
% ./labyrinth 6 4
6   4
1   2   2   8   3   4   4   10  5   6   5   11  6   12
7   13  8   9   8   14  9   15  10  11  12  18
13  19  14  20  15  21  16  17  16  22  17  23  18  24
19  20  21  22  23  24
%
```

Στο παραπάνω παράδειγμα, στην 1η γραμμή της εξόδου φαίνεται ότι έχουμε ένα λαβύρινθο με πλάτος 6 και ύψος 4 (είναι ο λαβύρινθος του σχήματος στην προηγούμενη σελίδα). Έπειτα ακολουθούν τα ζευγάρια των κελιών μεταξύ των οποίων οι δίοδοι είναι ανοικτές, δηλαδή το 1ο ζευγάρι, που είναι το 1 2, δηλώνει ότι μπορούμε να μεταβούμε από το κελί 1 στο κελί 2, καθώς και από το κελί 2 στο κελί 1. Το επόμενο ζευγάρι, που είναι το 2 8, δηλώνει ότι μπορούμε να μεταβούμε από το κελί 2 στο κελί 8, καθώς και από το 8 στο 2, κ.ο.κ.

Κάθε φορά που καλείται το πρόγραμμα `labyrinth`, παράγει και διαφορετικό λαβύρινθο, επειδή αρχικοποιεί τη γεννήτρια τυχαίων αριθμών με τον τρέχοντα χρόνο. Αν θέλετε να παράγετε τον ίδιο λαβύρινθο σε διαφορετικές εκτελέσεις του προγράμματος, μπορείτε να δώσετε σαν τρίτο όρισμα έναν αριθμό `<seed>` διάφορο του 0, ο οποίος θα χρησιμοποιηθεί σαν φύτρο για τη γεννήτρια. Συνεπώς, δίνοντας το ίδιο φύτρο (και την ίδια διάσταση), το πρόγραμμα παράγει τον ίδιο λαβύρινθο. Επίσης, δίνοντας σαν τέταρτο όρισμα έναν αριθμό `<draw>` διάφορο του 0, το πρόγραμμα θα παρουσιάσει τον λαβύρινθο και σχηματικά. Συνοψίζοντας, το πρόγραμμα `labyrinth` μπορεί να κληθεί ως

`labyrinth <width> <height> <seed> <draw>`

αλλά τα ορίσματα `<seed>` και `<draw>` είναι προαιρετικά. Παράδειγμα:

```
% ./labyrinth 6 4 9 1
6   4
1   2   1   7   2   8   3   4   3   9   4   5   5   6   5   11
7   13  8   14  10  11  10  16  11  12  12  18
13  19  14  20  15  16  15  21  17  18  17  23
20  21  22  23  23  24

+ +-+-+-+-+-+
|   |       |
+ + + +-+ +-+
| | | |     |
+ + +-+ +-+ +
| | |   |   |
+ + + +-+ +-+
| |   |     |
+-+-+-+-+-+ +
%
```

Το πρόγραμμα `escape`, που θα γράψετε, να διαβάζει από την είσοδο έναν λαβύρινθο στη μορφή που τον παράγει το πρόγραμμα `labyrinth` (όταν κληθεί χωρίς το όρισμα `<draw>`) και να εκτυπώνει στην έξοδο διαδοχικά τους αριθμούς των κελιών του μονοπατιού που οδηγεί από την είσοδο στην έξοδο του λαβυρίνθου. Παράδειγμα:

```
% ./labyrinth 6 4 9 > labyrinth.txt
% cat labyrinth.txt
6   4
1   2   1   7   2   8   3   4   3   9   4   5   5   6   5   11
7   13  8   14  10  11  10  16  11  12  12  18
13  19  14  20  15  16  15  21  17  18  17  23
20  21  22  23  23  24
% ./escape < labyrinth.txt
1 2 8 14 20 21 15 16 10 11 12 18 17 23 24
%
```

Αν κληθεί το πρόγραμμα escape και με το προαιρετικό όρισμα -draw, να εκτυπώνει τον λαβύρινθο σχηματικά εμφανίζοντας στο σχήμα και το μονοπάτι από την είσοδο στην έξοδο. Παράδειγμα:

```
% ./labyrinth 6 4 9 | ./escape -draw
1 2 8 14 20 21 15 16 10 11 12 18 17 23 24

+.+-+-+-+-+-+
|...|       |
+ +.+ +-+ +-+
| |.| |.....|
+ +.+-+.+-+.+
| |.|...|...|
+ +.+.+-+.+-+
| |...|  ...|
+-+-+-+-+-+.+
%
```

Ένα άλλο παράδειγμα:

```
% ./labyrinth 29 5 127 | ./escape -draw
1 30 31 2 3 32 33 62 61 90 91 92 93 64 65 66 95 96 97 98 99 70 71 42 41 40 39
38 37 8 9 10 11 12 13 14 15 44 43 72 101 102 73 74 45 16 17 18 47 48 19 20 21
50 49 78 77 76 75 104 105 134 135 136 107 108 137 138 109 80 81 82 83 112 141
142 143 144 145

+.+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|.|...  |   | |...............|.....|.....|   |           |
+.+.+.+-+ + + +.+-+-+-+-+-+-+.+.+ +.+.+-+.+ + +-+-+-+-+-+ +
|...|...| |   |...........|...|.| |...|...| |             |
+ +-+-+.+ +-+-+-+-+-+-+-+-+.+.+-+.+-+-+-+.+-+ +-+-+-+-+-+-+-+
|   |...| |.....      |...|.|...|.......| |.......|       |
+-+ +.+-+ +.+-+.+-+-+-+.+-+.+.+-+.+-+-+-+ +.+-+-+.+ +-+-+-+
|   |.......| |.........| |...| |...| |...|.| |  .|       |
+ +-+-+-+-+-+ +-+ +-+-+-+ +-+ + + +.+ +.+.+.+ + +.+-+-+-+ +
|           |           |     |   |.....|...|   |.........|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+.+
%
```

Είναι αποδοτικό το πρόγραμμα `escape` που γράψατε; Μπορεί να επιλύσει προβλήματα με μεγάλους λαβυρίνθους; Παράδειγμα:

```
% ./labyrinth 39 11 5 | ./escape -draw
1 40 79 80 119 118 157 158 197 236 237 198 159 160 199 238 277 276 275 314 353
354 393 394 395 356 357 358 359 360 361 400 401 402 363 362 323 284 283 244 243
204 165 166 205 206 167 168 129 128 89 90 51 12 13 14 53 54 15 16 55 56 17 18 57
96 135 136 175 174 213 214 253 292 291 290 289 288 327 326 325 364 365 404 405
366 367 406 407 408 409 410 371 370 369 368 329 330 331 332 293 254 215 176 137
138 177 216 217 256 257 258 219 220 259 298 337 338 377 378 379 380 419 420 421
382 381 342 343 344 345 346 347 386 385 424 425 426 427 428 429

+.+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|.      |   |     | |  .....|...|...|           |     |     |       |         |
+.+ +-+-+ + + +-+ + + +.+-+.+.+.+.+.+ +-+-+-+-+ +-+ + + +-+ + +-+-+-+ + + +-+-+
|.|   |   | | |   |   |.| |...|...|.|   | |   | |   |   |   | |       | | |   |
+.+-+ + +-+ + +-+-+ +-+.+ +-+-+-+ +.+-+ + + + + + +-+-+-+ +-+ + + +-+-+ + + + +
|...| |   | |   |   |...|     |   |.|   |   | | | |       |   | |   |   | | | |
+-+.+-+-+ + +-+ + +-+.+-+-+-+ + +-+.+ +-+-+-+ + + + +-+-+-+ +-+ +-+ + +-+ + + +
|...|   | |   |   | |...    | | | |...|...|   | | | |   |     | |   |   | | | |
+.+-+ +-+ +-+ +-+-+ +-+.+-+ + + + +-+.+.+.+ +-+ + + + + +-+-+ + +-+-+-+ + + + +
|...|...| |   | |...|...|   |   | |...|.|.  |   | | | | |     |   |   | |   | |
+-+.+.+.+ + +-+ +.+.+.+-+-+-+-+-+ +.+-+.+.+-+ +-+ + + + + +-+-+-+ + + +-+-+-+ +
|  .|.|.| |     |.|...|     |     |...|.|...| |...|   |   |     | | |   | |   |
+-+.+.+.+ + +-+ +.+-+-+ +-+ + +-+-+-+.+.+-+-+.+-+.+.+-+-+-+-+-+ +-+ + +-+ + + + +-+
|  ...|.| |   | |...|   |   |       |.|.| |.....|.|         |   |   | |   |   |
+ +-+-+.+-+ + +-+-+.+-+ + +-+ +-+-+-+.+.+ + +-+-+.+ +-+-+-+-+-+ +-+-+ + + +-+-+ +
| |.....|   | |   |...| |   |.........|.| |     |.|       |   |       | |   | |
+ +.+-+-+ +-+ + + +-+.+ +-+-+.+-+-+-+-+-+.+ +-+-+ +.+-+-+-+ + +-+-+-+-+-+-+ + + +
| |.|     |     |   |.  |.....|  .......| | |   |...|     |...........|   |   |
+ +.+-+ +-+-+-+-+-+-+-+.+-+-+.+-+-+-+.+-+-+-+ + + +-+ +.+-+-+-+.+-+-+-+-+.+ +-+-+-+
| |...| |...........|...|...|...|.......| |   |   |.......|...|   |...|       |
+ +-+.+ +.+-+-+-+-+.+-+.+ +.+.+.+-+-+-+-+.+ +-+-+ +-+-+-+-+-+.+-+.+ + +.+-+-+-+-+ +
|   |.....|       .....| |...|.........|               |.....| |  ...........|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+.+
%
```

Και κάποια θέματα προς διερεύνηση τώρα. Δεν είναι απαραίτητο ένας λαβύρινθος να έχει μία μόνο λύση, δηλαδή ακριβώς ένα μονοπάτι που συνδέει την είσοδο με την έξοδο. Μπορεί να μην υπάρχει λύση ή μπορεί να έχει περισσότερες της μίας.

Ο συγγραφέας του `labyrinth` ισχυρίζεται ότι οι λαβύρινθοι που παράγει το πρόγραμμα αυτό έχουν, εγγυημένα, ακριβώς ένα μονοπάτι που οδηγεί από την είσοδο στην έξοδο, αυτό που πρέπει να βρίσκει το πρόγραμμα `escape` που θα γράψετε. Μπορείτε να θεωρήσετε αληθή τον ισχυρισμό αυτό, αν σας διευκολύνει στην υλοποίηση του προγράμματός σας. Δεν θα ήταν άσχημα όμως, να καλύπτετε και την περίπτωση που δεν υπάρχει μονοπάτι που να είναι λύση του προβλήματος. Μπορείτε να πειραματιστείτε με αυτό το θέμα, παίρνοντας ένα λαβύρινθο που παρήχθη από το `labyrinth` και κλείνοντας μία δίοδο, με αφαίρεση του αντίστοιχου ζευγαριού κελιών, που βρίσκεται στο μονοπάτι που συνδέει την είσοδο με την έξοδο και δίνοντας το νέο λαβύρινθο σαν είσοδο στο πρόγραμμά σας.

Παράδειγμα:

```
% ./labyrinth 4 2 3 1
4   2
1   2   1   5   3   4   3   7   4   8
5   6   6   7

+ +-+-+-+
|   |   |
+ +-+ + +
|     | |
+-+-+-+ +
% ./labyrinth 4 2 3 | ./escape -draw
1 5 6 7 3 4 8

+.+-+-+-+
|.  |...|
+.+-+.+.+
|.....|.|
+-+-+-+.+
% echo 4 2 1 2 1 5 3 4 4 8 5 6 6 7 | ./escape -draw
Sorry, no way out!

+ +-+-+-+
|   |   |
+ +-+-+ +
|     | |
+-+-+-+ +
%
```

Υπάρχει φυσικά και το ενδεχόμενο λαβυρίνθων με περισσότερες της μίας λύσεις. Πώς συμπεριφέρεται το πρόγραμμά σας σ΄ αυτήν την περίπτωση; Δεν είναι απαραίτητο να χειριστείτε και αυτό το θέμα, θα ήταν καλό όμως να το σκεφτείτε και να τεκμηριώσετε στην αρχή του προγράμματός σας, με σχόλια, τι θα έπρεπε να διορθώσετε ώστε να βρίσκετε είτε μία από τις πολλές, είτε, ακόμα, και όλες τις λύσεις του προβλήματος. Αν θέλετε να πειραματιστείτε, μπορείτε να το κάνετε πάλι επεμβαίνοντας στο αποτέλεσμα του `labyrinth` με προσθήκες και επιπλέον διόδων που δημιουργούν και άλλα μονοπάτια που είναι λύσεις.

Για την άσκηση αυτή, θα παραδώσετε ένα μόνο αρχείο πηγαίου κώδικα C, το `escape.c`, και τίποτε άλλο. Οι οδηγίες υποβολής της άσκησης θα ανακοινωθούν έγκαιρα, μαζί με αυτές των ασκήσεων που θα ακολουθήσουν.