

ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ (2008-09)

Άσκηση 2

Πολλές φορές, έχουμε δεδομένα τα οποία είναι βολικό να εμφανίζονται στοιχισμένα σε στήλες. Για παράδειγμα, φανταστείτε ότι θα θέλαμε να είχαμε, στη μορφή ενός απλού κειμένου, έναν πίνακα με τα ονοματεπώνυμα των μελών ΔΕΠ του Τμήματος, τον τομέα που ανήκει το καθένα, το τηλέφωνό του, τον αριθμό του γραφείου του και την ηλεκτρονική του διεύθυνση, όπως φαίνεται στην ιστοσελίδα <http://www.di.uoa.gr/en/dep.php> του ιστοχώρου του Τμήματος, αλλά σε μία πιο απλή μορφή, σαν αυτή που βρίσκεται στην ιστοσελίδα <http://www.di.uoa.gr/~ip/hwfiles/dep.txt>. Παρατηρήστε τη δομή των γραμμών της ιστοσελίδας αυτής (το περιεχόμενό της παρατίθεται και στη συνέχεια) και προσέξτε, με τη βοήθεια των δύο γραμμών-οδηγών στην αρχή, τη θέση στην οποία ξεκινά κάθε στήλη του πίνακα.¹

```
0000000001111111112222222222333333333344444444445555555555666666666677777777778888888888
1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890
Apostolatos Nikolaos      1      5101      A31      napostol@di.uoa.gr
Filokyprou Georgios       2      5209      A14      gphilo@di.uoa.gr
Karoumbalos Konstantinos  3      5301      A42      kkarou@di.uoa.gr
Theofanous Nikiforos      3      5303      A54      ntheofa@di.uoa.gr
Delis Alex                 2      5212      A37      ad [AT]di.uoa.gr
Emiris Ioannis            1      5105      B20      emiris@di.uoa.gr
Georgiadis Panagiotis    2      5219      A23      p.georgiadis@di.uoa.gr
Halatsis Costas          2      5201      A27      halatsis@di.uoa.gr
Hatzopoulos Michalis     2      5203      B11      mike@di.uoa.gr
Ioannidis Yannis         2      5224      B12      yannis@di.uoa.gr
Kalouptsidis Nikos       3      5304      B22      kalou@di.uoa.gr
Koutsoupas Elias        1      5122      B7       elias@di.uoa.gr
Merakos Lazaros          3      5323      B15      merakos@di.uoa.gr
Misirlis Nikolaos        1      5103      I22      nmis@di.uoa.gr
Paschalis Antonis        2      5231      A39      paschali@di.uoa.gr
Sphicopoulos Thomas      3      5313      B17      thomas@di.uoa.gr
Stavarakis Ioannis       3      5343      A50      istavrak@di.uoa.gr
Syvridis Dimitris        3      5322,5335 A44      dsyvridi@di.uoa.gr
Theodoridis Sergios      3      5328      B21      stheodor@di.uoa.gr
Zissimopoulos Vassilios  1      5156      B55      vassilis@di.uoa.gr
Arapoyanni Aggeliki      3      5314      B19      arapoyanni@di.uoa.gr
Eleftheriades Alexandros 3      5210      A40      eleft@di.uoa.gr
Gounopoulos Dimitris     2      5227      B9       dg[AT]di.uoa.gr
Grigoriadou Maria        2      5205      A13      gregor@di.uoa.gr
Koubarakis Manolis       2      5213      B10      koubarak@di.uoa.gr
Manolakis Elias          3      5312      B23      eliasm@di.uoa.gr
Maroulis Dimitris        3      5307      A53      dmarou@di.uoa.gr
Martakos Drakoullis      2      5217      B1       martakos@di.uoa.gr
Mavrakis Dimitris        2      5732      A38      epgsec@kepa.uoa.gr
Rondogiannis Panagiotis  1      5102      B8       prondo@di.uoa.gr
Sagriotis Manolis        3      5310      A52      sagri@di.uoa.gr
Theoharis Theoharis      1      5106      I19      theotheo@di.uoa.gr
Cotronis Yiannis         2      5223      B2       cotronis@di.uoa.gr
Hadjiefthymiades Stathes 2      5148      B14      shadj@di.uoa.gr
Karali Isambo            2      5232      B4       izambo@di.uoa.gr
Karampogias Serafeim     3      5309      A43      mkara@di.uoa.gr
Kiayias Aggelos          1      -        -        -
Kolliopoulos Stavros     1      5108      B6       sgk@di.uoa.gr
Kouroupetroglou Georgios 3      5305      B13      koupe@di.uoa.gr
Molympakis Markos        3      5311      A55      molybaki@di.uoa.gr
Stamatopoulos Panagiotis 2      5202      A48      takis@di.uoa.gr
Stefanou Georgios        3      5308      A51      gstefa@di.uoa.gr
Tsalgatiou Aphrodite     2      5206      B2       atsalga@di.uoa.gr
Tzaferis Filippos        1      5112      I21      ftzaf@di.uoa.gr
Varoutas Dimitris        3      5318      B18      arkas@di.uoa.gr
```

¹Το ονοματεπώνυμο, ο τομέας, το τηλέφωνο, ο αριθμός γραφείου και η ηλεκτρονική διεύθυνση κάθε μέλους ΔΕΠ ξεκινούν από τις στήλες 1, 33, 41, 57 και 65, αντίστοιχα.

Μία άλλη περίπτωση που θέλουμε να έχουμε κάποιας μορφής στοίχιση είναι, όπως θα πρέπει να έχει γίνει σαφές μέχρι στιγμής στο μάθημα, στα προγράμματα που γράφουμε, και αυτό για λόγους καλύτερης αναγνωσιμότητάς τους. Για παράδειγμα, σε ένα πρόγραμμα C, οι εντολές μέσα στο σώμα μιας δομής επανάληψης (for, while ή do...while) πρέπει να είναι στοιχισμένες πιο δεξιά από τη στήλη στην οποία ξεκινά η εντολή της δομής επανάληψης. Το ίδιο ισχύει και για άλλες εντολές, όπως if, switch, κλπ. Δείτε, για παράδειγμα, τη διαφορά στην αναγνωσιμότητα των δύο προγραμμάτων http://www.di.uoa.gr/~ip/hwfiles/nonid_prog.c.txt και http://www.di.uoa.gr/~ip/hwfiles/id_prog.c.txt.

Το πρώτο πρόγραμμα είναι αυτό:

```
#include <stdio.h>
#define STARTYEAR 2001
#define ENDYEAR 2010

main()
{
int year, a, b, c, d, e;
for (year = STARTYEAR ; year <= ENDYEAR ; year++) {
a = year % 19;
b = year % 4;
c = year % 7;
d = (19*a+15) % 30;
e = (2*b+4*c+6*d+6) % 7;
printf("Easter in the year %d: ", year);
if (d+e+4 > 30)
printf(" May %d\n", d+e-26);
else
printf("April %d\n", d+e+4);
}
}
```

Ενώ το δεύτερο, εμφανώς πολύ πιο ευανάγνωστο από το πρώτο, είναι αυτό:

```
#include <stdio.h>
#define STARTYEAR 2001
#define ENDYEAR 2010

main()
{
int year, a, b, c, d, e;
for (year = STARTYEAR ; year <= ENDYEAR ; year++) {
a = year % 19;
b = year % 4;
c = year % 7;
d = (19*a+15) % 30;
e = (2*b+4*c+6*d+6) % 7;
printf("Easter in the year %d: ", year);
if (d+e+4 > 30)
printf(" May %d\n", d+e-26);
else
printf("April %d\n", d+e+4);
}
}
```

Ένας τρόπος για να επιτύχουμε την επιθυμητή στοίχιση σε δεδομένα μορφής πίνακα ή σε προγράμματα είναι να βάζουμε, για παράδειγμα στον κειμενογράφο που χρησιμοποιούμε για να τα συντάξουμε, συνεχόμενους κενούς χαρακτήρες. Όμως, ένας καλύτερος τρόπος είναι να χρησιμοποιούμε τον χαρακτήρα στηλογνώμονα (tab).² Κάθε τερματικό (hardware ή software) ή κάθε κειμενογράφος, που πρέπει να ερμηνεύσει στην έξοδο τον χαρακτήρα στηλογνώμονα, έχει εκ των προτέρων καθορισμένες τις λεγόμενες θέσεις στηλογνώμονα (tab stops). Συνήθως, οι θέσεις αυτές είναι οι 1, 9, 17, 25, 33, 41, ..., δηλαδή ανά 8 στήλες, αν και υπάρχουν τρόποι να αλλάξει αυτό, αν θέλει κάποιος. Η εκτύπωση ενός χαρακτήρα στηλογνώμονα έχει σαν αποτέλεσμα να μεταφερθεί η τρέχουσα θέση εκτύπωσης στην επόμενη θέση στηλογνώμονα, δηλαδή σαν να εκτυπωνόντουσαν κάποιοι κενοί χαρακτήρες μέχρι την επόμενη θέση στηλογνώμονα.

Αντικείμενο της άσκησης αυτής είναι να γράψετε δύο προγράμματα που θα διαβάζουν ένα κείμενο από την είσοδο και θα το μεταφέρουν στην έξοδο, έχοντας κάνει κάποιες τροποποιήσεις σε αυτό. Το πρώτο πρόγραμμα (έστω ότι το πηγαίο αρχείο του ονομάζεται `untabify.c`) θα μετατρέπει όλους τους χαρακτήρες στηλογνώμονα που θα διαβάζει από την είσοδο στον κατάλληλο αριθμό κενών χαρακτήρων, ώστε η έξοδος να μην φαίνεται διαφορετική από την είσοδο. Το δεύτερο πρόγραμμα (έστω ότι το πηγαίο αρχείο του ονομάζεται `tabify.c`) θα κάνει την αντίστροφη διαδικασία από το πρώτο. Δηλαδή, θα αντικαθιστά συνεχόμενους κενούς χαρακτήρες με χαρακτήρες στηλογνώμονα, όπου αυτό είναι δυνατόν να γίνει. Και στα δύο προγράμματα, ορίστε μία συμβολική σταθερά `TABDIST` με τιμή 8 (μέσω `#define`), η οποία καθορίζει τις αποστάσεις των θέσεων στηλογνώμονα. Σημειώνεται ότι σε ένα πρόγραμμα C, τον χαρακτήρα στηλογνώμονα τον αναπαριστούμε με `'\t'` και τον κενό χαρακτήρα με `' '`.

Ας δούμε κάποια αναλυτικά παραδείγματα. Για να μπορέσετε να τα αναπαραγάγετε και εσείς, κατεβάστε το αρχείο http://www.di.uoa.gr/~ip/hwfiles/tab_test.txt.³ Στα αποτελέσματα της εντολής `cat` στη συνέχεια, εμφανίζουμε με το σύμβολο `□` τους κενούς χαρακτήρες, με το `↑` τους χαρακτήρες στηλογνώμονα και με το `↓` της αλλαγές γραμμής. Εννοείται ότι αυτό γίνεται για λόγους καλύτερης κατανόησης του περιεχομένου των αρχείων. Όταν εκτελείτε τις εντολές αυτές σε ένα σύστημα, αυτό που φαίνεται στις θέσεις των `□` και `↑` είναι κενά διαστήματα.⁴

```
% cat tab_test.txt
```

```
+-----+-----+-----+-----+-----+-----+-----↓
ABCDE↑  FGHIJKLMN↑  ↑      □□OP□□↑ QRST↑  UVW↑  □□□XYZ□□↓
↑      □□abcde□fg□□□↑  □□□□□□□□□□□□hiijklmnopqrs↑  tuv□wxyz↓
```

```
% od -cv tab_test.txt
```

```
0000000  +  -  -  -  -  -  -  -  +  -  -  -  -  -  -  -
0000020  +  -  -  -  -  -  -  -  +  -  -  -  -  -  -  -
0000040  +  -  -  -  -  -  -  -  +  -  -  -  -  -  -  -
0000060  +  -  -  -  -  -  -  -  +  -  -  -  -  -  -  -
0000100  \n  A  B  C  D  E  \t  F  G  H  I  J  K  L  M  N
0000120  \t  \t                O  P                \t  Q  R  S  T  \t  U  V
```

²Ο χαρακτήρας αυτός αντιστοιχεί στο πλήκτρο που βρίσκεται συνήθως αριστερά από το Q στο πληκτρολόγιό μας και έχει, όπως και όλοι οι άλλοι χαρακτήρες, έναν ASCII κωδικό (είναι ο κωδικός 9, αλλά δεν έχει σημασία).

³Με την εντολή “od” του Unix μπορούμε να δούμε το “εσωτερικό” ενός αρχείου. Δώστε “man od” για περισσότερες πληροφορίες. Ο λόγος που χρησιμοποιούμε την εντολή αυτή στα παραδείγματα είναι για να διερευνήσουμε το ακριβές περιεχόμενο των αρχείων. Με μία απλή εκτύπωση, δεν είναι εμφανές αν κάπου στο αρχείο υπάρχει χαρακτήρας στηλογνώμονα ή ο ισοδύναμος αριθμός κενών χαρακτήρων. Αν δουλεύετε σε Windows και θέλετε να δείτε το “εσωτερικό” ενός αρχείου, κατεβάστε και εγκαταστήστε έναν hex editor, για παράδειγμα αυτόν που διατίθεται στη σελίδα <http://www.hhdsoftware.com/Products/home/hex-editor-free.html>.

⁴Η εντολή `diff` που φαίνεται στα παραδείγματα εμφανίζει τις γραμμές στις οποίες διαφέρουν δύο αρχεία. Αν τα αρχεία είναι πανομοιότυπα, δεν εμφανίζει τίποτα.

```

0000140  W \t          X Y Z          \n \t          a b
0000160  c d e          f g          \t
0000200          h i j k l m n o p q
0000220  r s \t t u v          w x y z \n
0000234

```

```
% ./untabify < tab_test.txt
```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----↓
ABCDEFGHIJKLMNOPQRSTUVWXYZ↓
abcdefghijklmnopqrstuvwxy↓

```

```
% ./untabify < tab_test.txt > tab_test_nt1.txt
```

```
% od -cv tab_test_nt1.txt
```

```

0000000  + - - - - - - - + - - - - - - - -
0000020  + - - - - - - - + - - - - - - - -
0000040  + - - - - - - - + - - - - - - - -
0000060  + - - - - - - - + - - - - - - - -
0000100  \n A B C D E          F G H I J K L
0000120  M N
0000140          O P          Q R S T
0000160          U V W          X Y Z
0000200  \n          a b c d
0000220  e          f g
0000240          h i j k l m n o p q
0000260  r s          t u v          w x
0000300  y z \n
0000303

```

```
% ./tabify < tab_test.txt
```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----↓
ABCDE↑ FGHIJKLMN↑ ↑      OP↑ QRST↑ UVW↑   XYZ↑ ↓
↑      abcde↑fg↑ ↑      hijklmnopqrs↑   tuv↓wxyz↓

```

```
% ./tabify < tab_test.txt > tab_test_yt1.txt
```

```
% od -cv tab_test_yt1.txt
```

```

0000000  + - - - - - - - + - - - - - - - -
0000020  + - - - - - - - + - - - - - - - -
0000040  + - - - - - - - + - - - - - - - -
0000060  + - - - - - - - + - - - - - - - -
0000100  \n A B C D E \t F G H I J K L M N
0000120  \t \t          O P \t Q R S T \t U V W \t
0000140          X Y Z \t \n \t          a b c d e
0000160          f g \t \t          h i j k l m n
0000200  o p q r s \t t u v          w x y z \n
0000217

```

```
% ./tabify < tab_test_nt1.txt > tab_test_yt2.txt
% ./untabify < tab_test_yt1.txt > tab_test_nt2.txt

% diff tab_test_nt1.txt tab_test_nt2.txt
% diff tab_test_yt1.txt tab_test_yt2.txt
%
```

Μία διευκρίνηση που πρέπει να γίνει για το πρόγραμμα `tabify.c` είναι η εξής: Πρέπει να γίνεται η μέγιστη δυνατή “συμπύεση” της εισόδου, μετατρέποντας συνεχόμενους κενούς χαρακτήρες σε χαρακτήρες στηλογνώμονα, όπου αυτό είναι δυνατόν. Δεν έχει όμως νόημα, και δεν πρέπει να γίνεται, η μετατροπή ενός μόνο κενού χαρακτήρα της εισόδου σε χαρακτήρα στηλογνώμονα, **εκτός και εάν μετά τη μετατροπή τον ακολουθούν και άλλοι χαρακτήρες στηλογνώμονα στην έξοδο**. Μελετήστε προσεκτικά, μέσω της εντολής `od`, τη δομή του αρχείου <http://www.di.uoa.gr/~ip/hwfiles/dep.txt> για να γίνει κατανοητή η απαίτηση που εκφράστηκε προηγουμένως. Το αρχείο αυτό έχει κατασκευασθεί από ένα πρόγραμμα `tabify.c` που ικανοποιεί τις παραπάνω προδιαγραφές, όταν του δόθηκε σαν είσοδος ένα αρχείο που περιείχε μόνο κενούς χαρακτήρες, αλλά κανένα χαρακτήρα στηλογνώμονα. Με άλλα λόγια, αν δώσετε σαν είσοδο το αρχείο αυτό στο `untabify.c` και μετά δώσετε το αποτέλεσμα της μετατροπής στο `tabify.c`, θα πρέπει να πάρετε πάλι ακριβώς το αρχικό αρχείο. Δηλαδή:

```
% ./untabify < dep.txt > dep_untabbed.txt
% ./tabify < dep_untabbed.txt > dep_tabbed.txt
% diff dep.txt dep_tabbed.txt
%
```

Ή σε μία εντολή:⁵

```
% ./untabify < dep.txt | ./tabify | cmp - dep.txt
%
```

Επαναλάβετε το ίδιο και για το αρχείο http://www.di.uoa.gr/~ip/hwfiles/id_prog.c.txt.

Η παράδοση της άσκησης αυτής συνίσταται στην υποβολή των πηγαίων αρχείων `untabify.c` και `tabify.c` με διαδικασία που θα ανακοινωθεί σύντομα.

⁵Κάντε “`man cmp`” για να δείτε τι κάνει η εντολή “`cmp`”.