

# Ασκήσεις Προετοιμασίας

Για την προετοιμασία σας για την 1<sup>η</sup> online εξέταση  
Σημείωση: Οι ασκήσεις αυτές ΔΕ βαθμολογούνται

## Θέμα 1. Εντολές Linux

- Τυπώστε στην οθόνη τα περιεχόμενα του αρχείου `stdio.h` που βρίσκεται στον κατάλογο `/usr/include`
- Τυπώστε στην οθόνη τις στήλες 3-6 του αρχείου `stdio.h` που βρίσκεται στον κατάλογο `/usr/include`
- Έστω το αρχείο `input.4col` το οποίο έχει πολλές γραμμές της μορφής:  
`StringA-StringB:Number1-Number2`  
Δημιουργήστε το αρχείο `output.4col` το οποίο έχει τα ίδια δεδομένα με το `input.4col`, αλλά έχει για διαχωριστικό το `tab` (`'\t'`) αντί για τα `':'` και `'-'`. Συνεπώς η μορφή του `output.4col` θα πρέπει να είναι:  
`StringA        StringB   Number1        Number2` (χωρισμένα με `tab`)
- Ταξινομήστε τις γραμμές του `output.4col` του προηγούμενου ερωτήματος ως προς την 3<sup>η</sup> στήλη (`Number1`) με αριθμητικό τρόπο, και μετά (σε περίπτωση ισοπαλίας) ως προς την 1<sup>η</sup> στήλη (`StringA`).
- Τυπώστε στην οθόνη τις 2 μικρότερες τιμές της 2<sup>ης</sup> στήλης (`StringB`) του αρχείου `input.4col`
- Τυπώστε στην οθόνη τις 2 μεγαλύτερες τιμές της 4<sup>ης</sup> στήλης (`Number2`) του αρχείου `input.4col`
- Τυπώστε στην οθόνη το 1<sup>ο</sup> (`StringA`) και το 3<sup>ο</sup> πεδίο (`Number1`) κάθε γραμμής του αρχείου `input.4col`
- Τυπώστε στην οθόνη μόνο τα δικαιώματα που έχετε σε αρχεία/φακέλους του `/usr/include`, καθώς και τα ονόματα των αρχείων/φακέλων αυτών.
- Βρείτε όλα τα αρχεία στο φάκελο `/usr/include` που το όνομά τους περιέχει το substring `"io"` και τυπώστε για το καθένα τέτοιο αρχείο την πρώτη γραμμή του.
- Δημιουργήστε με κάποια εντολή ένα κενό αρχείο `emptyFile`.
- Τροποποιήστε τα δικαιώματα του `emptyFile`.

**Οι εντολές θα πρέπει να εκτελούνται σε 1 μόνο γραμμή η κάθε μία. Επιτρέπεται να περιέχουν σωληνώσεις και ανακατευθύνσεις.**

## Θέμα 2. Πρόγραμμα findMaxColumn

Γράψτε ένα `bash` script πρόγραμμα **findMaxColumn** που να δέχεται ως παραμέτρους προγράμματος μία λίστα από αρχεία. Υποθέστε ότι κάθε αρχείο έχει γραμμές της μορφής:  
`num1:num2`

δηλαδή κάθε γραμμή περιέχει δύο ακέραιους αριθμούς (θετικούς ή αρνητικούς) που διαχωρίζονται από το χαρακτήρα `:`. Ο αριθμός γραμμών του κάθε αρχείου ΔΕΝ είναι σταθερός (οπότε κάπως θα πρέπει να τον ανακαλύψετε). Το μόνο σίγουρο είναι ότι κάθε αρχείο περιέχει τουλάχιστον 1 γραμμή.

Το ζητούμενο είναι να βρείτε τη στήλη (ανάμεσα σε όλα τα αρχεία εισόδου) με το μεγαλύτερο άθροισμα στοιχείων.

Πχ, αν κληθεί το πρόγραμμα ως:

**findMaxColumn ./input1 ../input2 input3**

ένα πιθανό αποτέλεσμα είναι:

FILE: input3, COLUMN: 2, sum=25

**ΣΗΜΕΙΩΣΗ: Το πρόγραμμά σας θα πρέπει στην εκτύπωση να ακολουθεί ΑΠΟΛΥΤΑ τη μορφή της παραπάνω εξόδου.**

Βοήθεια ανάγνωσης από αρχείο: <https://www.cyberciti.biz/faq/unix-howto-read-line-by-line-from-file/>

### Θέμα 3. Πρόγραμμα **printStats**

Γράψτε ένα bash script πρόγραμμα **printStats** σε Bash shell που να δέχεται ως παραμέτρους προγράμματος μία λίστα από αρχεία. Υποθέστε ότι κάθε αρχείο έχει σε κάθε γραμμή του ένα μόνο float αριθμό (πχ, -3.2), με το πολύ 1 δεκαδικό ψηφίο.

Το ζητούμενο είναι να τυπώσετε για κάθε αρχείο τον αριθμό των γραμμών του, το μέγιστο και το ελάχιστο στοιχείο που περιέχει.

Πχ, αν κληθεί το πρόγραμμα ως:

**printStats input1 input2 ./input3**

ένα πιθανό αποτέλεσμα είναι:

FILE: input1, lines: 8, max=25.3, min=-12.5

FILE: input2, lines: 3, max=21.1, min=2.0

FILE: ./input3, lines: 5, max=22.4, min=0.0

**ΣΗΜΕΙΩΣΗ:** Το πρόγραμμά σας θα πρέπει στην εκτύπωση να ακολουθεί **ΑΠΟΛΥΤΑ** τη μορφή της παραπάνω εξόδου.

Βοήθεια για την εκτύπωση: Ίσως χρειαστείτε την printf για ωραίο formatting:

<https://linuxconfig.org/bash-printf-syntax-basics-with-examples>

### Θέμα 4. Πρόγραμμα **printStats2**

Λύστε το θέμα 3 κάνοντας χρήση πινάκων (arrays). Ονομάστε το πρόγραμμά σας **printStats2**. Η είσοδος και η έξοδος του προγράμματος θα πρέπει να είναι ίδια με αυτή του προγράμματος **printStats**.