**Laborator 02**

**Setup infrastructură (Făcut în Laborator 01)**

* Verificați dacă folderul lab mai este mounted.
  + **mount | grep labs**
* Dacă nu este sau sunt probleme cu el:
  + **sudo pkill -9 sshfs**
  + **sudo umount /home/USER\_LOCAL/labs**
  + **sudo mount -a**
* Dacă vreți să verificați fișierele de pe server puteți folosi ssh sau WinSCP.
* Instalați [WinSCP](https://winscp.net/eng/index.php).
  + Când instalați, de la *User Interface Style* selectați *Commander*.
  + New Site
    - Host name: wiki.mta.ro
    - Port number: 30000
    - User name: cel de pe wiki.mta.ro
  + Advanced...
    - SSH >> Authentication >> Private Key File […] >> OK
  + Save >> Login
  + Stânga mergeți în directorul dorit - Dreapta folder-ul de pe server
  + Mergeți în folder-ul labs pe server.

**Exerciții**

**Pentru fiecare exercițiu se va scrie în fișierul \_REPORT.txt rezultatul rulărilor și răspunsul la întrebări.**

1. (**stackSize.c**) Doar pentru sisteme cu LINUX, pe Windows mărimea stivei este de 1MB. Scrieți cod în C care să afișeze dimensiunea stivei (Căutați pe Google apel system() și ulimit -a)
2. (**bigVector.c**) Declarați un vector de caractere numit **a,** de mărime **N,** cu **N** fiind un număr definit cu #define.
   * Declararea lui **a** se va face în interiorul funcției **main()**.
   * Umpleți acest vector cu 0-uri și afișați ultimul element.
   * Testați programul anterior cu **N** setat la 100 și cu **N** setat la o valoare mai mare decât mărimea stack-ului.
   * Declarați un pointer la un șir de caractere **b** și alocați pentru acesta 1GB de memorie.
   * Repetați alocarea precedentă de un milion de ori. Adăugați verificare, când pointerul este **NULL** afișați numărul iterației la care s-a ajuns.
   * Cât spațiu RAM ați putut aloca?
3. (**sizeof.c**) Analizați diferența dintre diferitele metode de a aloca vectori:
   * Alocați un vector de 1000 de caractere, inițializat cu șirul “Hello World!”.
   * Alocați un pointer la un șir de caractere inițializat cu șirul “Hello World!”.
   * Alocați un pointer de caractere neinițializat pentru care alocați dinamic 1000 de caractere.
   * Afișați pentru toate 3 mărimea datelor (sizeof) și mărimea șirurilor de caractere.
   * Afișați cei 3 pointeri și adresele lor.
   * Scrieți în **\_REPORT.txt** ce observați din afișări.
4. (**complex.c**) Scrieți un program pentru lucru cu numere complexe.
   * Se definește o structură ce reprezintă un număr complex (două float-uri).
   * Se definesc ca funcții operațiile de bază: adunare, scădere, înmulțire.
   * Se implementează cod de evidențiere a funcționalității celor 3.
5. (**allStruct.c**) Definiți o structură ce conține **toate** tipurile de date pe care le cunoașteți.

**Exercițiile de la 1 la 5** sunt **obligatorii**.

1. (**3dMatrix.c**) Declarați o matrice tridimensională, inițializați toate valorile sale cu 1 și implementați funcția care calculează suma tuturor valorilor sale.