

Tema 1 aplica?ie  
client-server

Cerin?e generale:

2. Metoda de procesare a datelor pe server trebuie s? fie f?cut? concurrent? -

utiliza?i rutine. 3. Exist? un fi?ier de configurare în care exist? parametrii ini?iali ai programului.

Ex: câte elemente are array-ul de date pe care îl poate trimite clientul, de câte ori se apeleaz? sau rutin? go. Decide?i voi ce anume pute?i include în acest fi?ier.

? Client <Nume> Conectat. ? Client <Nume> a facut request cu date: <date>. ? Server a primit requestul. ? Server proceseaza datele. ? Server trimite <raspuns> c?tre client. ? Client <Nume> a primit r?spunsul: <raspuns>. Cerin?e individuale:

Serverul returneaz? c?tre client un array cu cuvinte unde cuvântul i din lista output este alcatuit din caracterele de pe pozi?ia i în fiecare din cuvintele din array-ul input.

Exemplu: casa, masa, trei, tanc, 4321 => cmtt4, aara3, ssen2, aaic1 Pentru pozitia 0 avem cmtt4 pentru ca sunt alese in ordinea caracterelor de pozitia 0 din fiecare string, deci c din casa, m din masa etc. 2. Clientul trimite catre server un array de strings . Un string poate con?ine atât

caractere, cât ?i cifre, amestecate. Serverul returneaz? c?tre num?rul clientului de numere care sunt p?trate perfecte. Exemplu: abd4g5, 1sdf6fd, fd2fdsf5 => 2 p?trate perfecte: 16 din 1sdf6fd, 25 dinfd2fdsf5.

Serverul de r?spundere c?tre client cu suma numerelor format prin inversarea fiec?rui element din array-ul ini?ial. Exemplu: 12, 13, 14 => 21, 31, 41 cu suma 93

4. Clientul trimite c?tre server un array având pe primele două poziții limitate întregi a, b

ale unui interval și un șir de valori întregi. Serverul returnează media aritmetică a numerelor citite, pentru care suma cifrelor aparține intervalului [a,b]. Exemplu:

Pentru a = 2, b = 10, n = 5 și valorile 11, 39, 32, 80, 84 server-ul va returna 41. 5.

Clientul trimite c?tre server un array de strings. .

Serverul elimina toate stringurile ce nu reprezintă numere binare, iar pentru cele ce reprezintă numere binare va face conversia în baza 10 și va returna clientului.

Exemplu: 2dasdas, 12, dasdas, 1010, 101 => 10, 3 6. Clientul trimite c?tre server un array de strings. Un string poate conține doar

caractere. Serverul return c?tre client codificarea Caesar ([https://en.wikipedia.org/wiki/Caesar\\_cipher](https://en.wikipedia.org/wiki/Caesar_cipher)) a element elementare, use LEFT sau RIGHT shift cu k poziții. Exemplu: abcdef => xyzabc folosit LEFT shift cu 3 caractere. abcdef => bcdefg utilizeaza RIGHT shift cu 1 caracter. 7. Clientul trimite c?tre server un text codificat după regulamentul următor: în fața respectiv?

caracterul este scris un număr ce număr de apariții consecutive al acestuia. Server-ul returnează textul decodificat. Numărul ce apare în fața unui caracter va fi mai mic sau cel mult egal cu 20. Exemplu: Pentru ?1G11o1L? se va afișa ?Goooooooooooo L?. 8. Clientul trimite c?tre server un array de numere naturale .

Server-ul returnează numărul total de cifre ale tuturor numerelor prime din șir.

Exemplu: Pentru: 23, 17, 15, 3, 18 => 5 cifre (nr 23, 17, 3) 9. Clientul trimite c?tre server un array de strings. Un string poate conține atât

caractere, cât și cifre, amestecate. Serverul răspunde c?tre client cu numărul de cuvinte care au un număr de cuvinte aflate pe poziții par în cuvânt. Exemplu: mama, iris, bunica, ala => 2 cuvinte: iris, ala. 10. Clientul trimite c?tre server un array de strings. Un string poate conține atât

caractere, cât și cifre, amestecate. Serverul returnează cmmdc pentru toate numerele (se calculează listele divizorilor pentru fiecare număr și se calculează intersecția listelor). Exemplu: 24, 16, 8, aaa, bbb => 8.

Serverul de răspunde c?tre client cu suma numerelor array-ului format prin permutarea la dreapta a k cifre ale fiecărui element din array-ul inițial.

Exemplu: 1234, 3456, 4567 => pentru  $k = 2$ :  $3412 + 5634 + 6745 = 15791$ . 12.

Clientul trimite c?tre server un array de numere naturale .

Serverul returneaz? c?tre client suma elementelor array-ului format din dublarea primei cifre a num?rului. Exemplu: 23, 43, 26, 74 => suma de returnat =  $223 + 443 + 226 + 774$ . 13.

Clientul trimite c?tre server un array de ?ntregi ce sunt pe primele dou? pozi?ii limitate

?ntregi  $a, b$  ale unui interval, restul array-ului reprezent?nd componentele a  $n$  numere complexe (elementele din array luate dou? c?te c?te dou? imagini din partea real? ?i cea a unui num?r complex). Server-ul returneaz? ?n ordine cresc?toare valorile modulelor ce nu apar?in intervalului  $[a, b]$ . - a se folosi pachetul `cmplx` (<https://golang.org/pkg/math/cmplx/#Abs> ). Exemplu: 3,10,3,4,5,6: intervalul  $[3,10]$  ?i numerele complexe:  $3+4i, 5+6i$ .

Serverul returneaz? c?tre un array de stringuri cu toate stringurile ar putea s? fie parole valide (contin litere mici, mari, cifre ?i simboluri). Puteti folosi regex.

Exemplu: Ceva12!@, asa212, dasdas => Ceva12!@ 15. Clientul trimite c?tre server un array de caractere(pot sa fie strings de

dimensiune 1) ce pot sa fie litere mici, mari, cifre sau simboluri . Serverul returneaz? c?tre client un num?r  $N$  de parole,  $N$  fiind aleator ales ?ntre 1 ?i 10. Fiecare cuv?nt este de dimensiune  $M$ , unde  $M$  este ?ntre 5 ?i 15 ?i este alc?tuit? din caracterele primite de la client, av?nd minim o litera mica, o litera mare, un simbol si o cifra. Exemplu: a, b, e, 3, !, A => b A3ba, ee3!Bab3, aabb A! Serverul intoarce 3 parole (3 ales aleator din intervalul 1-10) de dimensiune 5, 8, respectiv 6 (alese aleator din intervalul 5-15). Barem: 1.5p 1. Aplica?ie cu un singur client-server 0.2p 2. Mai mul?i clien?i: 0.5p 3. Citire din fi?ier: 0.2p 4. Mesaje client-server: 0.2p 5. Arhitectura general? a aplica?iei: 0.4p 6. Termen de prezentare

o. 20.11.2025 (Grupa 1) b.

20.11.2025 (Grupa 2)