

Cerință

Se va crea un program Python de tip „*subnet calculator*” care să prelucreze folosind funcțiile de CGI datele dintr-un formular HTML și să întoarcă rezultatele pe o pagină de web.

Formularul va avea următoarele câmpuri:

- o adresă IP, sub forma xxx.xxx.xxx.xxx

- o lungime de mască IP (*netmask*), sub forma unui număr N (echivalent cu masca /N; de exemplu masca /25 înseamnă *netmask=255.255.255.128*)

- un buton Submit

Pentru echipele cu numărul de forma 2k, rețeaua se va împărți în 8 subrețele; pt. 2k+1, în 4 subrețele.

La apăsarea butonului Submit, programul va verifica faptul că adresa și masca sînt valide, în caz contrar raportînd un mesaj de eroare.

În caz afirmativ, el va împărți rețeaua dată de masca /N (ignorînd partea de host) în nr. dat de subrețele și va afișa, pentru fiecare subrețea creată, adresa de rețea, netmask-ul, hosturile disponibile și adresa de broadcast. Se vor folosi **culori diferite** (la alegere) pt a afișa octeții de **host** și de **broadcast**.

Exemplu:

IP=192.168.1.174, N=24, echipa 1 →4 subrețele

Rețea = 192.168.1.0/24, se ignoră ultimii 8 biți (hostul .174)

Subrețele:

192.168.0.0/26, netmask 255.255.255.192, hosturi 192.168.0.1-62, broadcast 192.168.0.63

192.168.0.64/26, netmask 255.255.255.192, hosturi 192.168.0.65-126, broadcast 192.168.0.127

192.168.0.128/26, netmask 255.255.255.192, hosturi 192.168.0.129-190, broadcast 192.168.0.191

192.168.0.192/26, netmask 255.255.255.192, hosturi 192.168.0.193-254, broadcast 192.168.0.255

Orice detaliu necesar funcționării și care nu a fost specificat mai sus este la alegerea voastră, și-l veți comenta în fișierul sursă și/sau în *readme.txt*.

Tema este bazată pe lucrarea de lab. „HTTP și CGI”, care trebuie studiată în prealabil pt. explicațiile respective (inclusiv exemplele furnizate în directorul /asi/cgi-bin de pe matrix.elcom.pub.ro). Dacă programul python se termină anormal, atunci cînd este rulat ca un cgi sub serverul de web, acesta va afișa pe pagină mesajul generic: *Error 500 – the server made a boo boo.*

Permisuni și restricții

Pagina web a echipei N se va numi *tema3.htm* și se vizualizează pe server la următorul URL:

<http://matrix.elcom.pub.ro/~asiN/tema3.htm>

Semnificația caracterului ~ este „home directory”. Serverul este configurat a.î. fișierele .htm și .py trebuie să se afle în folderul /home/asi/asiN/**public_html** (locație standard în UNIX). Nu puteți să afișați conținut HTML și să rulați scripturi cgi aparținînd utilizatorului, din afara lui *public_html*. Avantajul este că puteți pune/edita acolo fișiere cu permisiuni proprii, nu de *root*.

Atenție! Serverul permite doar ca fișierele .py să fie executabile ca scripturi cgi. Dacă au alte extensii nu vor funcționa. În plus, fișierele .py tb să aibă atributul executabil (+x). Setați acest atribut folosind „chmod a+x fisier.py”. Nu lăsați fișierele ne-executabile, ca la alte lucrări, urmînd să fie apelate sub forma „python fisier.py” întrucît serverul de HTTP nu funcționează așa!

Mod de predare

În directorul personal al echipei curente *~asiN/public_html* se vor găsi cele 2 fișiere (html și Python), pentru a rula conform cerinței. Programul trebuie rulat prin intermediul paginii de web date, fiind apelat prin formularul HTML. Nu se va rula nimic din linia de comandă (puteți face acest lucru doar pentru teste).

Veți include un fișier *tema3/readme.txt* care va descrie pe scurt modul de operare și funcționare a programelor.

Mod de testare - notare

- voi citi explicațiile din *readme.txt*

- voi verifica funcționarea cf. specificațiilor, la adresa de web <http://matrix.elcom.pub.ro/~asiN/tema3.htm>

- voi examina sumar sursele, urmărind cel puțin să nu fie surse identice pentru echipe diferite (nerespectarea celor două cerințe de mai sus va duce la obținerea notei 1, respectiv 0)