



Informatii utile:

Enunt + intrare/iesire: <http://www.netromsoftware.ro/uploads/180110/STL2018.zip>
Solutiile se trimit la: catalin@netrom.ro

Introducere

Intr-o aplicatie de tip ERP (Enterprise Resource Planning) se doreste implementarea unui sistem de permisiuni, prin care diferitii utilizatori ai sistemului (din diferite departamente, cu diferite functii) au acces, sau au restrictionat accesul, la functionalitatile sistemului. Pentru a acoperi gama larga de elemente ale sistemului care au nevoie de un acces controlat, dar si pentru a permite o definire cat mai facila a lor, se defineste un sistem de permisiuni, bazat pe urmatoarele concepte:

- Utilizator: Un utilizator al sistemului, autentificat pe baza unui username unic si a unei chei personale (parole);
- Rol: O functie in cadrul aplicatiei, ce presupune accesul la anumite functionalitati ale sistemului;
- Permisune: un element al aplicatiei (un screen, un feature, un buton, etc) al carui acces se face diferentiat in functie de permisiunile utilizatorului curent;
- Nivel de acces: nivelul de acces asociat unei permisiuni, reprezentat ca o reuniune dintre valorile urmatoare: read (r), write (w), execute (x), deny (d)

Enunt

Dandu-se un set de utilizatori, rolurile asociate lor, precum si nivelul de acces al unei permisiuni asociata unui rol, scrieti un algoritm care determina urmatoarele:

- Pentru un utilizator si o permisiune date, care este nivelul de acces rezultat din toate rolurile care sunt asociate utilizatorului;
- Care sunt utilizatorii „administrator” din sistem, adica acei utilizatori care au toate nivelele de acces (read, write si execute) pentru toate permisiunile existente in sistem.

Observatii:

- Un utilizator poate avea mai multe roluri asociate.
- Un rol poate avea asociate mai multe nivele de acces pentru aceeasi permisiune.
- Nivelul de acces al unui utilizator pentru o anumita permisiune se determina prin reuniunea nivelelor de acces ale respectivei permisiuni asociate cu rolurile asignate acestui utilizator. In cazul in care nivelul de acces pentru permisiune este deny in unul din rolurile asociate, acesta invalideaza toate celelalte nivele de acces asociate permisiunii in celelalte roluri ale utilizatorului, nivelul de acces final fiind deny (d).

Descriere intrare

Inputul este citit dintr-un fisier de intrare "input.txt" sub forma urmatoare:

- Pe prima linie: n (numarul de asocieri intre utilizatori si roluri)
- Pe liniile 2 pana la (n+1), asocieri de utilizatori si roluri, in formatul <user> <rol>
- Pe linia (n+2): p (numarul de asocieri intre roluri si permisiuni)
- Pe liniile (n+3) pana la (n+p+2), asocieri de roluri si permisiuni, in formatul <rol> <permisiune> <nivel de acces>, unde nivelul de acces este un string ce poate contine unul sau mai multe nivele de acces in ordinea r(ead), w(rite), e(xecute), d(eny). E.g. 'rwx', 'r', 'd', etc

- Pe linia (n+p+3): u (numarul de utilizatori pentru care se doreste determinarea nivelului de acces asociat unei permisiuni)
- Pe liniile (n+p+4) pana la (n+p+u+3), utilizatorii si permisiunile pentru care se doreste determinarea nivelului de acces, in fomatul <user> <permisiune>

Observatii:

- Se considera ca utilizatorii din sistem sunt toti cei care au macar o asociere cu un rol.
- Se considera ca rolurile din sistem sunt toate rolurile care au macar o asociere cu un utilizator.
- Se considera ca permisiunile sistemului sunt toate permisiunile care exista macar o data asociate cu un rol.
- O permisiune poate sa fie asociata de mai multe ori cu un rol, cu acelasi nivel de acces sau cu unul diferit.

Descriere iesire

Programul trebuie sa afiseze in consola (sau intr-un fisier de output) nivelul de acces pentru fiecare asociere <user><permisiune> prezenta in fisierul de input intre liniile (n+p+4) si (n+p+u+3) in formatul <user> <nivel de acces>. Formatul nivelului de acces este pe patru caractere 'rwx-', acolo unde nivelul de acces lipseste afisandu-se caracterul '-' (e.g. 're--', '---d'). Apoi programul trebuie sa afiseze numele utilizatorilor care sunt "administrator", adica au nivel de acces complet (rwx-) pentru toate permisiunile definite in sistem, in forma: "<user> is admin".

Exemplu

Input

```
10
user1 role1
user1 role2
user1 role4
user2 role2
user2 role3
user3 role1
user4 role1
user4 role2
user4 role3
user5 role3
9
role1 p1 rw
role1 p2 r
role1 p2 w
role2 p2 x
role2 p3 r
role3 p1 x
role3 p3 w
role3 p3 x
role4 p1 d
4
user1 p3
user1 p1
user4 p2
user5 p2
```

Output

user1 p3 r---
user1 p1 ---d
user4 p2 rwx-
user5 p2 ----
user4 is admin

Explicatie

user1

user1 are rolurile role1, role2 si role4

Determinare nivel de acces p1

Asociata cu role1, permisiunea p1 are nivelul de acces *r*, respectiv *w*

Permisiunea p1 nu este asociata cu rolul role2

Asociata cu role4, permisiunea p1 are nivelul de acces *d* (*deny*)

Concluzie: Pentru ca *d* are exclusivitate, rezulta ca utilizatorul user1 are permisiunea deny pentru permisiunea p1, deci outputul corect este: user1 p1 ---d

Determinare nivel de acces p3

Permisiunea p3 nu este asociata cu rolul role1

Asociata cu role2, permisiunea p3 are nivelul de acces *r*

Permisiunea p3 nu este asociata cu rolul role4

Concluzie: nivelul de acces al utilizatorului user1 pentru permisiunea p3, este r---

user4

user4 are rolurile role1, role2 si role3

Determinare nivel de acces p2

Asociata cu role1, permisiunea p2 are nivelul de acces *r*, respectiv *w*

Asociata cu role2, permisiunea p2 are nivelul de acces *x*

Permisiunea p2 nu este asociata cu rolul role4

Concluzie: user4 are pentru permisiunea p2, nivelul de acces: rwx-

user5

user5 are rolul role3

Determinare nivel de acces p2

Permisiunea p2 nu este asociata cu rolul role3

Concluzie: user5 are pentru permisiunea p2, nivelul de acces: ----

Determinarea administratorilor

Aplicatia defineste 3 permisiuni: p1, p2, p3

Utilizatorul user4 are pentru permisiunea p1, nivelul de acces: rwx-

Utilizatorul user4 are pentru permisiunea p2, nivelul de acces: rwx-

Utilizatorul user4 are pentru permisiunea p3, nivelul de acces: rwx-

Concluzie: utilizatorul user4 este administrator