

WUOLAH



Juancaruiife

www.wuolah.com/student/Juancaruiife



2033

Ej3-Clase-RedSocial.pdf

Enero 2019 + Soluciones



1º Fundamentos de Programación



Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada

```
1 class RedSocial{
2 private:
3     static const int MAXIMO_USUARIO = 100;
4     string Usuarios[MAXIMO_USUARIO];
5     bool RelacionesAmistad[MAXIMO_USUARIO][MAXIMO_USUARIO];
6     int usuarios_utiles;
7
8 public:
9     void HacerAmigos(string usr1, string usr2);
10    void DeshacerAmigos(string usr1, string usr2);
11    bool SonAmigos(string usr1, string usr2);
12    bool Existe(string usr);
13
14    int getIndice(string user){
15        for (int i = 0; i < usuarios_utiles; i++)
16        {
17            if(user == Usuarios[i]) return i;
18        }
19    }
20
21    string getNombre(int indice){
22        return Usuarios[indice];
23    }
24
25    string Amigo_Potencial(string user){
26        int usuario = getIndice(user),
27            cnt = 0,
28            ContadorAmigosComunes[usuarios_utiles],
29            mayor = 0,
30            indice = 0;
31        for (int i = 0; i < usuarios_utiles; i++) //recorremos todos los usuarios, excepto el que vamos a averiguar su amigoPotencial.
32        {
33            if (i != user && !SonAmigos(user, getNombre(i)))
34            {
35                for (int j = 0; j < usuarios_utiles; j++) //recorremos la matriz de amigos del usuario que queramos comparar
36                {
37                    if (SonAmigos(getNombre(i),getNombre(j))) //eliminamos la posibilidad de que no sean amigos
38                    {
39                        if (SonAmigos(user, getNombre(j))) // si se cumple todo lo anterior, el usuario aportado tendra un amigo comun mas con este usuario
40                            cnt++;
41                    }
42                }
43                ContadorAmigosComunes[i] = cnt; // almacenamos con el indice de los usuarios, cuantos amigos comunes tiene con ese "i" usuario
44            }
45        }
46
47        for (int i = 0; i < usuarios_utiles; i++) //ahora vamos a averiguar cual es el que tiene mas amigos en comun
48        {
```

```
49         if(mayor < ContadorAmigosComunes[i]){
50             mayor = ContadorAmigosComunes[i];
51             indice = i;
52         }
53     }
54     return getNombre(indice);
55 }
56
57 bool amigos_circunstanciales(string A, string B){
58     int i_a = getIndice(A),
59         i_b = getIndice(B),
60         cnt = 0;
61
62     if (SonAmigos(A,B)) //Comprobamos que sean amigos para evitar errores, ↗
63         en los exámenes siempre van a pillar en eso
64     {
65         for(int i=0; i < usuarios_utiles; i++){
66             if(SonAmigos(A, getNombre(i)) && i != i_b) return false;
67             if(SonAmigos(B, getNombre(i)) && i != i_a) return false;
68         }
69         return true; //Devolvera verdadero si y solo si no tienen ningun ↗
70         amigo mas que ellos, es decir, el anterior bucle for termina sin ↗
71         interrupciones
72     }
73     else return false;
74 }
75 };
```