

Problema 16

SOTILLO SOTILLO, CLARA (TAIS96)

ID envío	Usuario/a	Hora envío	Veredicto
57789	TAIS96	2022-10-05 10:29	AC
57766	TAIS96	2022-10-05 10:12	AC
57753	TAIS96	2022-10-05 10:06	WA
57751	TAIS96	2022-10-05 10:06	CE

Fichero ejercicio16.cpp

*

* Indicad el nombre completo y usuario del juez de quienes habéis hecho esta solución:

* Estudiante 1: Carlos Gómez Lopez TAIS45

* Estudiante 2: Clara Sotillo Sotillo TAIS96

*

El coste en el caso peor de la función es $O(N+M)$ siendo M el numero de aristas del grafo y N el numero de vertices de este.

la razon de este coste es porque se recorre una vez todos los vertices del grafo y porque recorre el numero de aristas 2 veces, la primera cuando es la primera vez que pasa por los vertices

y la segunda cuando otro vertice pasa por esa arista aunque los vertices adyacentes ya hayan sido visitados, pero en vez de poner $2M$ se pone M porque es constante.

Creamos un vector que guarde el dinero que cuesta que cada personaje empiece a difundir el rumor

Creamos un grafo con cada uno de los personajes (N) como vertices y como aristas los pares de amigos que hay (M)

Declaramos una clase busqueda, a la que le pasamos como parametro un grafo y el vector del dinero

su funcionalidad es obtener la suma de la menor cantidad de oro que tiene cada componente conexa del grafo

para ello realizamos una busqueda en profundidad la cual recorre cada componente conexa comparandola con el minimo establecido.

```
class busqueda {
```

```
public:
```

```
    busqueda(Grafo const& g, vector<int> const& vec) : visit(g.V(), false), dist(g.V(), 0) {
        for (int i = 0; i < g.V(); i++) {
            if (!visit[i]) {
                min = vec[i];
                dfs(g, i, vec);
                dinero += min;
            }
        }
    }
```

```

    }
}
int din() {
    return dinero;
}
private:
    vector<bool> visit;
    vector<int> dist;
    int min;
    int dinero = 0;

    void dfs(Grafo const& G, int v, vector<int> const& vec) {
        visit[v] = true;

        for (int w : G.ady(v)) {
            if (!visit[w]) {

                if (vec[w] < min) {
                    min = vec[w];
                }

                dfs(G, w, vec);
            }
        }
    }
};

```



```

bool resuelveCaso() {

    // leemos la entrada
    int N, M, aux, aux2;
    vector<int> v;
    cin >> N >> M;

    if (!cin)
        return false;

    for (int i = 0; i < N; i++) {
        cin >> aux;
        v.push_back(aux);
    }

    Grafo g(N);

    for (int i = 0; i < M; i++) {
        cin >> aux >> aux2;
        g.ponArista(aux-1, aux2-1);
    }
}

```

```
    busqueda b(g, v);  
  
    cout << b.din();  
  
    cout << "\n";  
    return true;  
}
```