

# Networking

Progettare ed implementare un programma che consenta di gestire le informazioni relative a un sistema per il networking professionale. Tutte le classi, ad eccezione della classe `Esempio`, devono appartenere al package `networking`.

## R1. Network

La classe principale con cui interagire è la classe `Network`, il cui costruttore riceve come parametri la regione di competenza della rete, un codice univoco identificativo della rete, ed il numero massimo di utenti che la piattaforma è in grado di gestire (ad esempio, la piattaforma di networking `TO-1`, operativa sulla regione `Piemonte`, con un numero massimo di 100 utenti memorizzabili). Tali informazioni sono accessibili attraverso i rispettivi metodi `getCodice()`, `getRegione()` e `getNumeroMassimoUtenti()`.

Ogni piattaforma di networking offre un numero massimo di collegamenti tra utenti sulla base del costo di abbonamento. Attraverso il metodo `setNumeroMassimoCollegamenti()` è possibile impostare il numero massimo di collegamenti che ogni utente registrato in una determinata piattaforma potrà avere. Il metodo `descrizioneNetwork()` della classe `Network` restituisce una stringa contenente il codice identificativo della piattaforma, la regione di operatività, il numero massimo di utenti gestibili e il numero massimo di collegamenti tra utenti, separati da una virgola e da uno spazio: ad esempio `TO-1, Piemonte, 100, 5` per una piattaforma che può gestire 100 utenti e che consente un massimo di 5 collegamenti per ogni utente.

## R2. Utenti

Il programma memorizza le informazioni relative agli utenti gestiti da una determinata piattaforma. L'aggiunta di un utente ad una piattaforma avviene attraverso il metodo `nuovoUtente()` della classe `Network`, che riceve come parametri il codice fiscale del nuovo utente, il nome, il cognome e una stringa che ne descrive l'occupazione; il codice fiscale rappresenta l'identificativo univoco di ogni utente. Il sistema controlla che all'interno della stessa piattaforma non siano registrati più utenti con tale codice: cercando di aggiungere un utente già presente (stesso codice fiscale), il metodo aggiorna le informazioni di tale utente. In caso di avvenuta creazione o aggiornamento di un utente, il metodo restituisce una stringa contenente le informazioni su tale utente separate da virgola e spazio; ad esempio, `GRCFBA93P17L219V, Fabio, Garcea, Dottorando`. Cercando invece di aggiungere un numero di utenti superiore a quello ammissibile per la piattaforma, il metodo non sortisce alcun effetto (ed il valore di ritorno non è rilevante). Si consideri invece, che utenti con lo stesso codice fiscale possono essere registrati su piattaforme diverse (ad esempio su diverse regioni).

Per ottenere le informazioni relative ad un utente registrato su una determinata piattaforma di networking è possibile utilizzare il metodo `utente()` della classe `Network`, che riceve come parametro il codice fiscale dell'utente e restituisce una stringa contenente le sue informazioni nello stesso formato del metodo precedente. In caso di utente inesistente il metodo non sortisce alcun effetto (ed il valore di ritorno non è rilevante).

Il metodo `utenti()` della classe `Network` restituisce invece un array contenente i codici fiscali degli utenti memorizzati su una determinata piattaforma. La dimensione dell'array è pari al numero di utenti attualmente registrati e i codici fiscali sono riportati in ordine di aggiunta nel sistema.

## R3. Collegamenti

La piattaforma di networking permette agli utenti di entrare in contatto e di formare una rete di comunicazione professionale. Per creare un collegamento tra utenti registrati su una stessa piattaforma di networking si utilizza il metodo `nuovoCollegamento()` della classe `Network`. Il metodo riceve come parametri due codici fiscali e una data che rappresenta il giorno di creazione del collegamento; nel caso in cui entrambi i codici fiscali fossero legati a utenti effettivamente memorizzati nella piattaforma, il metodo genera il collegamento tra di essi e assegna al collegamento un codice numerico incrementale (a partire da 0). In caso di avvenuta creazione del collegamento, il metodo restituisce una stringa riportante il codice identificativo del collegamento e la data separati dal carattere ``-'`. Ad esempio, `0-12/10/2021`. Nel caso in cui uno o più degli utenti non esistano, o nel caso in cui uno di essi abbia raggiunto il limite massimo di collegamenti consentiti, il metodo non sortisce alcun effetto (ed il valore di ritorno non è rilevante).

Per ottenere le informazioni relative ad un collegamento è possibile utilizzare il metodo `collegamento()` della classe `Network`, che riceve come parametri due codici fiscali di due utenti. Nel caso di utenti effettivamente registrati e di collegamento esistente, il metodo restituisce il valore `true`. In tutti gli altri, casi il metodo non sortisce alcun effetto ed il valore di ritorno è `false`.

Il metodo `collegamenti()` della classe `Network` restituisce una stringa contenente tutti i collegamenti finora creati, uno per riga, nell'ordine in cui sono stati aggiunti. Ogni riga ha il formato sopra riportato, ed è terminata da un carattere di 'a capo' (tranne l'ultima). Attraverso il metodo `collegamentiPerOccupazione()` è possibile ottenere l'elenco dei collegamenti in cui almeno uno degli utenti coinvolti svolge l'occupazione ricevuta come parametro, nel formato utilizzato dal metodo precedente.

