

CC1002-3**Profesor:** Patricio Inostroza**Auxiliares:** Juan José Alegría

e Isidora Araya

Fecha: Miércoles 14 de

Diciembre.

Auxiliar 14

1. Amiguitos (P1 C3 2011)

Para representar la información de los amigos (como en facebook) se puede utilizar un diccionario que a cada persona le asocia sus amigos.

Por ejemplo, $D = \text{"gabi":}[\text{"jose"}, \text{"rosa"}]$, $\text{"jose":}[\text{"gabi"}]$, $\text{"rosa":}[\text{"mati"}, \text{"gabi"}]$, Para su manejo se dispone de la siguiente clase:

```
class Amigos:

    def __init__(self, archivo): #inicializa diccionario con información grabada en archivo
        self.D={} #diccionario inicialmente vacío
        for linea in open(archivo, "r"):
            amigos=linea.split("/") #lista con persona y sus amigos. Ej: ["gabi", "jose", "rosa"]
            persona=amigos[0]; self.D[persona]=[]
            for amigo in amigos[1:len(amigos)]:
                self.D[persona].append(amigo)

    def amigar(self, X, Y):... #registrar que X es amigo de Y y que Y es amigo de X

    def grabarAmigos(self, archivo):... #grabar el diccionario en el archivo nombrado
```

Las personas que solicitan ser amigos de X se encuentran en una línea del archivo "X.txt". Por ej, "mati.txt" contiene "jose/ gabi".

La información completa está (y debe quedar) grabada en el archivo "amigos.txt". Cada línea del archivo contiene el nombre y los amigos separados por /. Por ejemplo: "gabi/jose/rosa" indica que los amigos de gabi son jose y rosa.

A partir de lo anterior, realice lo siguiente:

- Escribir los métodos amigar y grabarAmigos
- Use la clase anterior en un programa que le permita a una persona procesar las solicitudes de amistad recibidas en la forma:

Cuál es tu nombre?mati

mati, quieres ser amigo(a) de jose (si/no)?si

mati, quieres ser amigo(a) de gabi (si/no)?no

...