

CC1002 Introducción a la Programación

Auxiliar 9

Prof. Benjamin Bustos
Auxs. Cristobal Sepúlveda, Franco Sepúlveda

Fecha: 18 de octubre de 2017

Resolver los problemas usando Python, utilizando estrictamente la Receta de Diseño entregada a lo largo del curso. Use nombres apropiados para funciones y variables, y realice testing cada vez que sea posible.

1. Listas de Python

Los nombres de los amigos(as) de una persona están guardados en la lista de strings `amigos`. Por ejemplo:

```
amigos = ['Juan', 'Ana', 'Rosa', 'Jose', 'Nico']
```

Suponga que en la lista no hay nombres repetidos.

- i. Sea `invitado` una estructura mutable definida como

```
estructura.mutable('invitado', 'nombre confirmado')}
```

Escriba un procedimiento que se invoque en la forma `invitar(amigos)`, que reciba una lista de Strings con los nombres de amigos de una persona, y produzca el siguiente efecto: modifique la variable de estado `invitados` (lista de `invitado`) con los amigos(as) que la persona desea invitar a una fiesta. El procedimiento debe establecer el diálogo indicado en el siguiente ejemplo:

```
Invita a Juan (si o no)?no
Invita a Ana (si o no)?si
Invita a Rosa (si o no)?si
Invita a Jose (si o no)?no
Invita a Nico (si o no)?si
```

El procedimiento debe modificar la lista `invitados`, agregando los nombres de las personas invitadas y valores *False* (de tipo *bool*) para el atributo `confirmado`. Para el ejemplo, la invocación `invitar(amigos)` modificaría la variable de estado a lo siguiente:

```
[invitado('Ana', False), invitado('Rosa', False), invitado('Nico', False)]
```

- ii. Implemente el procedimiento **confirmar**, que recibe una lista de strings con los nombres de las personas que confirmaron su asistencia, y modifique la lista de invitados (variable de estado) marcando **True** en la componente **confirmado**. Siguiendo con el ejemplo del punto anterior, **confirmar**(['Nico', 'Rosa']) modifica la lista **invitados** y queda como sigue:

```
[invitado('Ana', False), invitado('Rosa', True), invitado('Nico', True)]
```

2. Fecha para comer

Un grupo de compañeros de trabajo quiere fijar una fecha para reunirse a comer durante el próximo mes. Como pueden ser muchos compañeros y cada uno de ellos tiene ya comprometidas un buen número de fechas de dicho mes, se pide desarrollar un programa que indique las posibles fechas cuando realizar la comida.

Cada compañero de trabajo escribirá las fechas que tiene comprometidas (es decir, que no puede usar) como una secuencia de enteros finalizada con un número negativo, ordenadas ascendentemente menos la última (el número negativo). Es decir, habrá tantos comensales como números negativos. Para finalizar de introducirse fechas comprometidas, se ingresa el valor cero (marca de finalización). En ese momento, el programa deberá mostrar en pantalla las posibles fechas para realizar la comida (las fechas en donde todos pueden asistir).