

Algoritmos y Programación I

75.40

Trabajo práctico N°2

Facultad de Ingeniería, Universidad de Buenos Aires

Segundo cuatrimestre del 2016

Alumnos: Silvestri, María Carolina – Del Mazo, Federico

Legajo: 99175 - 100029

Práctica: Alan

E-mail: silconito@gmail.com - delmazofederico@gmail.com

Correctora: Ana Czarnitzki

Planteo del problema

El problema propuesto por la práctica consiste en realizar un programa que almacene notas. Este programa debe poder agregar notas, listarlas (dentro del programa y en versión HTML), buscar entre ellas y eliminarlas, según lo que el usuario desee.

Resolución y funciones

El tronco principal del programa consistirá en, primero, preguntarle al usuario que archivo desea manejar (con el fin de que distintos usuarios puedan guardar distintos archivos, o el mismo usuario distintas notas) y luego un menú que le mostrara por pantalla todas las funciones que tiene el programa. En el menú también el usuario debe poder cambiar el archivo o salir del programa. Es importante notar que hay que diferenciar las funciones que modifican un archivo de las funciones que simplemente interactúan con el usuario y también que debe haber funciones auxiliares a las que las funciones principales puedan llamar en todo momento.

Funciones auxiliares

Las dos funciones auxiliares, a las que las funciones principales del programa estarán llamando constantemente son:

- Una función que lea todo el archivo CSV y cree un diccionario con su contenido. Este diccionario será dinámico y espontáneo, generándose en el momento que cada función lo llame en vez de convertirse en una constante que estaría siendo acarreado por el programa en todo momento. El diccionario contiene la clave, que es la etiqueta y su valor asociado: una lista con cadenas, es decir las notas asociadas a la etiqueta
- Una función que confirme que el archivo CSV existe y tiene notas ya escritas, con el fin de frenar a las funciones de eliminar o buscar en ese archivo, ya que no habría nada para eliminar/buscar/etcétera

Agregar notas

La opción de agregar notas está compuesta por dos distintas funciones. Por un lado está la función que pide al usuario la nota y la etiqueta a agregar, esta función guarda ese par de valores y de ahí se lo pasa a la función que verdaderamente escribe las notas al archivo.

Esta segunda función es la única que puede crear el archivo CSV, ya que se abre en modo 'a'. De modo que si el nombre del archivo no existe lo crea, y, cuando sí existe, la nota va a ser agregada al final de la línea.

Listar

La función listar es de las más simples del programa. Esta es una función que no interactúa con el usuario ni abre ni cierra el archivo, simplemente imprime las etiquetas y las notas. La función listar requiere de las dos funciones auxiliares que están detalladas abajo.

Buscar

El usuario al ingresar cualquier cadena de caracteres que desea buscar recibirá un listado de las notas que contienen esa cadena. Estas notas estarán impresas y agrupadas por etiquetas, la función no devuelve ningún valor.

Por comodidad, la función busca directamente sobre el diccionario dinámico del archivo, en vez del archivo en sí, ahorrando así el abrir el archivo.

Eliminar

La función eliminar, al igual que la de agregar, se divide en dos partes, una para interactuar con el usuario y la otra para eliminar las notas seleccionadas por este.

La primera función de las dos le da a elegir de manera interactiva al usuario (mostrándole las etiquetas del archivo, enumerando las notas, etcétera) que nota desea eliminar y devuelve ese valor.

La segunda función, luego de una confirmación para corroborar que el usuario no se haya equivocado, procede a abrir el archivo en modo lectura, guardarse en una variable nueva todas las líneas del archivo excepto la seleccionada, y

luego escribe esa variable en el archivo, con este abierto en modo escritura. Se eligió esta forma de hacerlo porque el modo de escritura trunca el archivo, entonces en vez de usar algún método muy rebuscado de borrar una línea en particular, es mas sencillo abrirlo, guardar todo y reescribirlo desde cero.

Generar versión HTML

Esta opción utilizara nuevamente dos funciones distintas. En la primera, haciendo uso del método *insert* de las listas de Python, se le agregará las etiquetas y notas a una lista pre formateada con el formato del HTML. Luego, esta lista HTML modificada la agarra la segunda función que simplemente crea un nuevo archivo con extensión HTML y escribe esas líneas en el archivo.

Este archivo HTML creado listará las notas con sus etiquetas, tal cual lo hace el modo listar del programa, solo que ahora lo tenemos en un archivo “físico” en vez de en una opción de un programa.