

Práctica 1

Vilchis Domínguez Miguel Alonso

1. Contexto:

La hermana menor de tu amigo Alan está aprendiendo las operaciones básicas de matemáticas, como sumar, restar, multiplicar y dividir. Él le está ayudando en ese difícil proceso, sin embargo, no puede garantizar que ella no hace trampa debido a que ambos tienen acceso a la computadora de escritorio.

Por lo que te pidió que escribieras un programa, que funcionara como una calculadora, pero que antes de poder usarla pidiera usuario y contraseña, de tal manera que él podría asegurarse que ella no utilizó la calculadora para resolver los ejercicios que le son dejados.

2. Objetivo:

Que el alumno se familiarice con el lenguaje de programación Python y que aprenda a implementar interfaces gráficas.

3. Especificaciones:

- El alumno puede crear su GUI con cualquier biblioteca que desee, aunque se recomienda usar PyQt4
- Debe de crear 4 vistas:

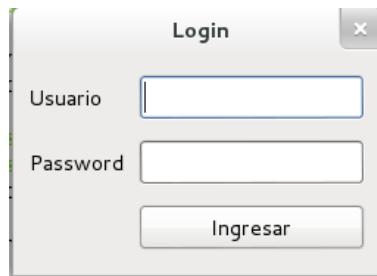


Figura 1: Imagen que muestra la etapa de Login

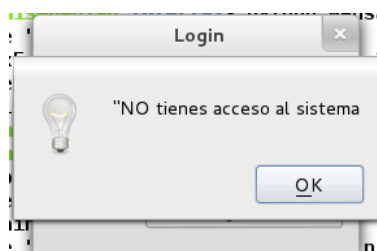


Figura 2: Imagen que muestra Error de Ingreso

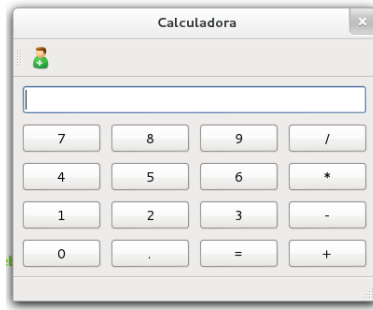


Figura 3: Imagen que muestra éxito de Ingreso, por lo tanto, mostrar la calculadora. (La calculadora debe tener la funcionalidad de añadir usuario)

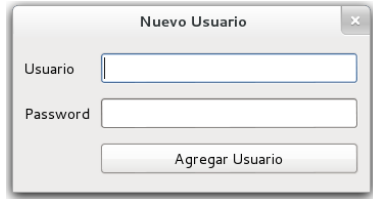


Figura 4: Imagen que muestra la opción de añadir usuario

- La “base de datos.” estará almacenada en un archivo llamado input.txt. Por default, ésta contará con un usuario:

username : root password original : root

- Por temor a que la hermana menor descubra el archivo, por cada caracter del password se tomará su valor ascii y se le sumará 5 unidades, el caracter que esté representado por el nuevo valor ascii será el que se almacenará en el archivo. Deberás tomar las precauciones necesarias para reconocer el password original.
- Deberás crear 2 calculadoras, una simple que solo sume y reste y una más compleja (heredará de la primera) que incluya suma, resta, multiplicación, división (cuyo resultado será de punto flotante aunque los operandos sean enteros), módulo y potencia. Ambas trabajarán con operaciones binarias, del estilo: $4+3$ o $10 * 10000$, o $54/23.3$ o $3.2 - 32$, no hace falta implementar paréntesis u operaciones más complejas. (La interfaz que mostrarán será de la calculadora compleja)
- La estructura de la práctica deberá ser dividida de la siguiente manera:

```

|
| - --Constants
| - --GUI
| - --Code
|
| - --Main.py
| - --Constants.py
| - --*.py
| - --Calculator.py
| - --ScientificCalculator.py
| - --Input.txt

```

- En la carpeta de Constants deberán declarar todas las constantes que usen en el proyecto, no derán aparecer números mágicos.
- En la carpeta de GUI debe ir el código relacionado con la interfaz gráfica.
- En la carpeta Code va lo relacionado al código de la calculadora, (sin interfaz gráfica) y el archivo de contraseñas.
- Al ejecutar `$python Main.py` se debe de desplegar su ventana de login.

4. Entrega

Fecha de entrega: 15 de Agosto del 2016

La entrega es por parejas, la práctica con la estructura mencionada debe estar almacenada en un depósito de github llamado Redes2017, en la rama Practica1, el readme debe contener el nombre de los 2 integrantes del equipo.

Deben mandar el link del depósito con el asunto *[Redes-17]Practica1* al siguiente correo:

mvilchis@ciencias.unam.mx

Nota: Basta con que un integrante del equipo suba su código al depósito.

Nota: A las 11:59 del día de la entrega se descargará el contenido de los depósitos, para evaluación, asegúrense que se encuentre su versión final para esa hora.