

Proyecto de programación

INTRODUCCIÓN

Esta tarea tiene como propósito producir una pieza de software de forma colaborativa a lo largo de un período extendido de tiempo. A diferencia de los ejercicios desarrollados en clase, controles y evaluaciones formativas, requiere una mayor planificación y coordinación entre los integrantes del equipo de desarrollo, lo que es más cercano a las condiciones de trabajo de un profesional de la informática.

El trabajo será realizado en grupos compuestos por 3 o 4 estudiantes, conformados libremente. Estos deben ser **comunicados a los profesores hasta el día 22 de octubre**, vía correo electrónico.

OBJETIVO

Se deberá crear un programa graficador que genere tres tipos de gráficos (barras, torta y línea). El programa debe preguntar al usuario el tipo de gráfico a desplegar y la paleta de colores a utilizar. Tanto los datos del gráfico como la definición de los colores deben leerse desde archivos. Los gráficos deben generarse utilizando Turtle.

FORMATO DE LOS ARCHIVOS DE ENTRADA

El programa deberá cargar los datos que se graficarán desde un archivo CSV¹ con dos columnas. En la primera aparecerá el nombre del ítem y en la segunda el valor correspondiente. El archivo tendrá un número indeterminado de filas. Un ejemplo de archivo se muestra a continuación. Note que en este ejemplo aparecen 4 ítems, pero podrían ser 1 o más. El programa debe verificar no solo que el archivo exista, sino también que tenga el formato requerido.

Chile,295
Perú,205
Brasil,580
Colombia,283

¹ https://es.wikipedia.org/wiki/Valores_separados_por_comas

Los gráficos se colorearán de acuerdo a una paleta de colores, la cual estará especificada en un archivo que puede contener una o más paletas. El formato de archivo de la paleta de colores se compone de una cantidad indeterminada de pares de líneas. La primera línea de cada par, inicializada por un carácter ">" indica el nombre de la paleta, y la línea siguiente indica en formato hexadecimal la combinación RGB de una cantidad indeterminada de colores. Los gráficos deben colorearse asignando los colores en orden. Si el gráfico tuviera más ítems que la cantidad de colores que tiene la paleta, ésta deberá repetirse para el siguiente ítem. Un ejemplo de archivo de paleta de colores se muestra a continuación. Note que la cantidad de colores varía: la primera paleta tiene 3 colores mientras que la segunda tiene 4. Las paletas pueden tener 1 o más colores. Al igual que para el archivo de datos, el programa debe verificar no solo que el archivo de colores exista, sino también que tenga el formato requerido.

```
>colores primarios  
ffff00,0000ff,ff0000  
>tonos azules  
cccccc,00ffff,389eb8,0000ff
```

INTERACCIÓN

Al ejecutarse, el programa deberá preguntar por el nombre del archivo desde donde se recogerán los datos, ofrecer tres opciones de gráfico (barras, torta y línea), ofrecer tantas opciones de paletas de colores como contenga el archivo de paletas y pedir un nombre para el gráfico.

El programa debe verificar que los archivos de datos y de paleta ingresados por el usuario existan y que tengan el formato correcto. En caso contrario, debe informarlo al usuario y preguntar nuevamente. Los ingresos del usuario deben validarse, evitando que el programa se caiga.

A continuación se muestran cuatro ejemplos de interacción, con sus correspondientes salidas. Tanto la interacción como los gráficos deben ser lo más similares a estos ejemplos.

```
$ python3 graficador.py
Ingrese nombre del archivo de datos: empresas_segurida.dat
  No existe un archivo con ese nombre, inténtelo nuevamente
Ingrese nombre del archivo de datos: empresas.dat
  El archivo no tiene el formato requerido, inténtelo nuevamente
Ingrese nombre del archivo de datos: empresas_seguridad.dat

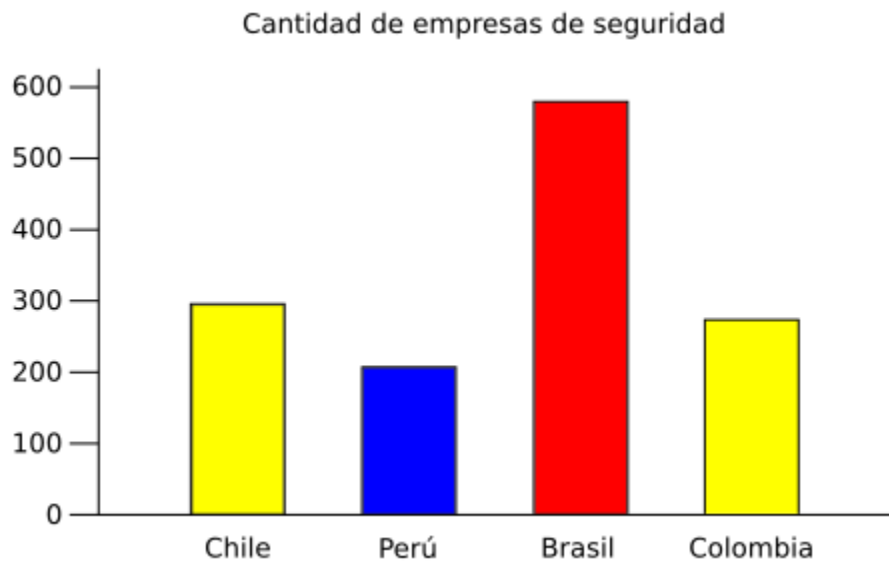
Ingrese nombre del archivo de colores: colores.dat

Seleccione paleta de colores para su gráfico:
  1) colores primarios
  2) tonos azules
Ingrese opción: 3
  Opción no disponible
Ingrese opción: 1

Seleccione el tipo de gráfico:
  1) barras
  2) torta
  3) línea
Ingrese opción: 1

Ingrese un título para el gráfico: Cantidad de empresas de seguridad

generando gráfico...
```



```
$ python3 graficador.py
$ python3 graficador.py
Ingrese nombre del archivo de datos: empresas_seguridad.dat

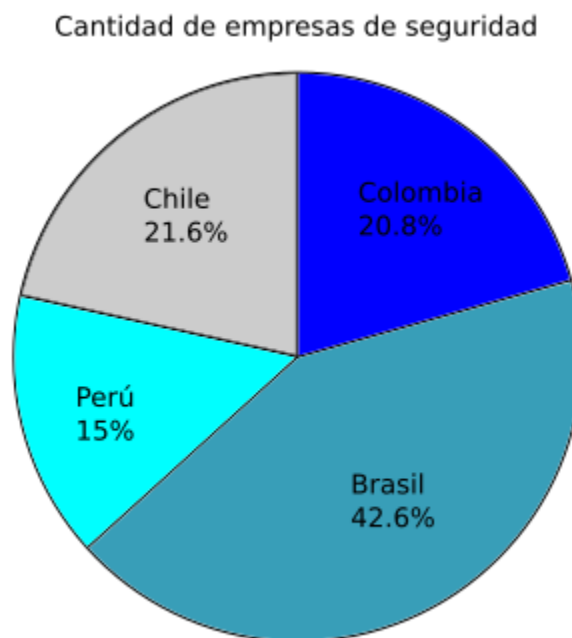
Ingrese nombre del archivo de colores: kolores.dat
  El archivo no tiene el formato requerido
  Ingrese un nuevo nombre o presione ENTER para abandonar el programa
Ingrese nombre del archivo de colores: colores.dat

Seleccione paleta de colores para su gráfico:
  1) colores primarios
  2) tonos azules
Ingrese opción: 2

Seleccione el tipo de gráfico:
  1) barras
  2) torta
  3) línea
Ingrese opción: 2

Ingrese un título para el gráfico: Cantidad de empresas de seguridad

generando gráfico...
```



```
$ python3 graficador.py
$ python3 graficador.py
Ingrese nombre del archivo de datos: empresas_seguridad.dat

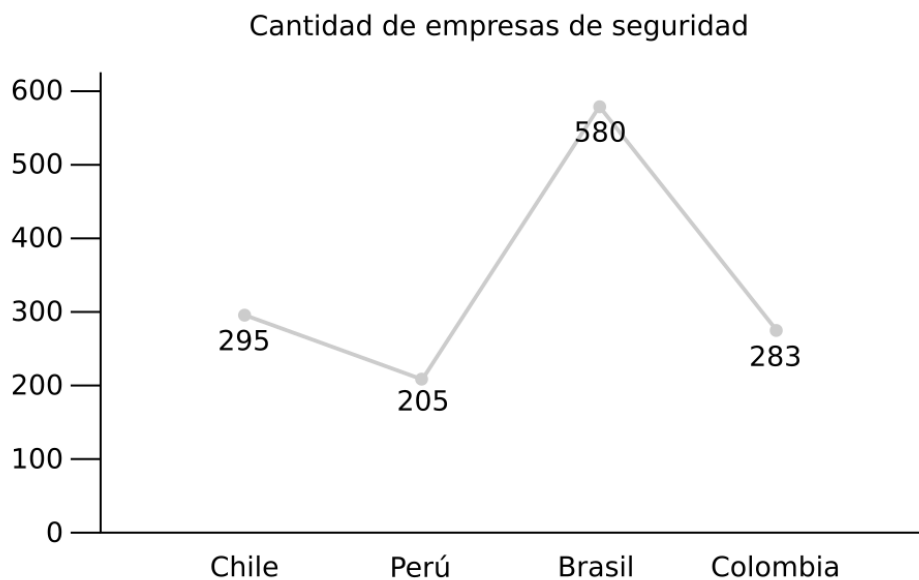
Ingrese nombre del archivo de colores: paletaColores.dat
No existe un archivo con ese nombre
Ingrese un nuevo nombre o presione ENTER para abandonar el programa
Ingrese nombre del archivo de colores: colores.dat

Seleccione paleta de colores para su gráfico:
  1) colores primarios
  2) tonos azules
Ingrese opción: 2

Seleccione el tipo de gráfico:
  1) barras
  2) torta
  3) línea
Ingrese opción: 2

Ingrese un título para el gráfico: Cantidad de empresas de seguridad

generando gráfico...
```



Note que en el caso del gráfico de líneas, la línea y los puntos son coloreados del primer color de la paleta seleccionada.

```
$ python3 graficador.py
$ python3 graficador.py
Ingrese nombre del archivo de datos: empresas_seguridad.dat

Ingrese nombre del archivo de colores: paletaColores.dat
  No existe un archivo con ese nombre
  Ingrese un nuevo nombre o presione ENTER para abandonar el programa
Ingrese nombre del archivo de colores:

Programa finalizado.

$
```

ASPECTOS DE PROGRAMACIÓN

Hasta este punto se ha descrito el comportamiento externo del programa. Es importante (y se incluye en la evaluación), que esté programado de una buena forma. Entre otras cosas, deben cuidarse los siguientes aspectos:

- **Código legible:** buen espaciamiento, comentarios documentando funciones, nombres de variables y funciones apropiados.
- **Código simple:** evitar ciclos de más, código duplicado, condiciones que nunca se cumplen, etc.
- **Estructura:** que el código sea modular: se estructure en funciones con un propósito bien definido, bien comentadas y bien utilizadas por las funciones que la invocan, debe existir una función main() que organice el flujo principal del programa.
- **Uso de contenedores:** buena elección de contenedor (diccionario, lista, set, etc.) de acuerdo al problema que se esté resolviendo.
- **Uso de métodos:** validación de errores mediante excepciones, buena elección de ciclos (for o while), etc.
- **Otros aspectos.**

INFORME

En conjunto con el código, deberá entregarse un informe en formato PDF, con una longitud no superior a 3 páginas, que recoja su reflexión sobre la experiencia ganada durante la realización del proyecto.

La estructura del informe es la siguiente:

- **Encabezado:** Título del informe, identificación de integrantes e identificación de curso: $\frac{1}{3}$ de página
- **División del trabajo:** descripción del trabajo realizado por cada integrante: $\frac{2}{3}$ de página
- **Descripción de la forma de trabajo:** cómo decidieron la estructura del programa, cómo organizaron el trabajo, cómo integraron el trabajo de cada integrante. 1 página
- **Aprendizajes para el futuro:** qué prácticas (organización, programación, comunicación entre integrantes u otras) resultaron bien y sería bueno repetir, así como qué dificultades se presentaron que podrían preverse en el futuro. 1 página.

Pese a lo breve del informe, se espera que este recoja información de calidad. Se recomienda ir tomando nota de las cosas que van funcionando y las que no a medida que se avanza en el trabajo. Este aprendizaje permitirá mejorar sus prácticas de programación y contribuirá a ser mejores profesionales en el futuro.

ENTREGA

Se deberá entregar una carpeta comprimida que contenga el archivo .py, un archivo de datos, uno de colores y el informe en PDF vía **Siveduc** hasta el día **14 de noviembre a las 23:59 hrs.** Se aplicará una penalización de 1 punto por cada hora de retraso. Se recomienda planificar el trabajo para ser entregado 24 horas antes de la fecha de cierre.

EVALUACIÓN

Se probará el programa con distintos archivos de datos y paletas de colores no necesariamente los entregados por los estudiantes. Procure que su programa recoja los datos de estos archivos y despliegue limpiamente la información en los gráficos. Podrán realizarse cambios menores en el despliegue en casos de necesidad (por ejemplo, en el gráfico de barra se pueden poner los nombres de los ítem en posición vertical si son muchos, o en el gráfico de torta ponerlos fuera de la torta. En cualquier caso, la información mostrada en el gráfico debe ser comprensible.

La evaluación se desglosará de la siguiente forma:

- 1 punto: correcta lectura y validación de archivos
- 1 punto: correcto despliegue del gráfico de barras
- 1 punto: correcto despliegue del gráfico de torta
- 1 punto: correcto despliegue del gráfico de líneas
- 1 punto: aspectos de programación.
- 1 punto: informe