

FACULTAD DE INGENIERÍA
FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA
PRUEBA PARCIAL PROGRAMADA N°2



INSTRUCCIONES

Antes de comenzar, lea atentamente las instrucciones, los requerimientos de entrega y el problema

- La evaluación es de carácter individual. Cualquier indicio de intervención de otra persona será calificado con nota mínima a la evaluación y quedará impedido de rendir cualquier instancia de evaluación optativa.
- En caso de detectar programas iguales o con suficientes similitudes entre sí, tanto en una misma sección como en secciones distintas, se calificará con la nota mínima a todos los involucrados y quedarán impedidos de rendir cualquier instancia de evaluación optativa. Para ello se compararán las entregas con un algoritmo de reconocimiento de texto y se revisarán los casos sospechosos mediante un proceso de validación visual que realizará el cuerpo docente del curso.
- Estudiantes que entreguen fuera de plazo serán calificados con nota mínima en esta evaluación.
- Entregas que se realicen por vías distintas a Campus Virtual serán calificadas con nota mínima. En caso de que la plataforma presente un problema, puede enviar su archivo como respaldo al correo de contacto del profesor de Teoría, y posteriormente subir el archivo a Campus Virtual.
- Estudiantes que entreguen sin seguir los criterios de identificación estipulados en el apartado de "entrega" serán calificados con nota mínima en esta evaluación.
- La difusión de este enunciado solicitando ayuda, ofreciendo dinero a cambio de la resolución e incluso la publicación de éste dentro del plazo de desarrollo de la tarea, se considera un acto deshonesto, que daña la imagen de la Facultad de Ingeniería y de la Universidad de Santiago de Chile, y significará la reprobación inmediata de la asignatura y la aplicación de los procedimientos disciplinarios que el reglamento de ética de la Universidad dispone para estos casos.
- Considere que cualquier supuesto que el estudiante haga debe ser explicitado en los comentarios de código.
- La subida del archivo es responsabilidad de su autor, por lo que, archivos que no estén en el formato estipulado, que vengan corruptos o con problemas para ser leídos, no serán revisados.
- Cualquier situación no contemplada en este documento será dirimida por la Facultad de Ingeniería.

ENTREGA

Se recibirán soluciones hasta las 23:59 del día 15 de enero de 2022, en la plataforma Campus Virtual, en el espacio habilitado para ello en la pestaña “Evaluaciones -> Pruebas”.

Se requiere entregar un único archivo .py con la solución al problema, el archivo debe llevar por nombre el RUT de su autor (sin puntos ni guión) por ejemplo:

19984321K.py

Además, al inicio del archivo, se debe añadir el siguiente encabezado del programa, con los datos solicitados para identificar su trabajo (Rellene el encabezado del programa con SUS DATOS PERSONALES en RESPETANDO EL MISMO FORMATO al indicado en el ejemplo del Código a continuación):

```
# FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA
# SECCIÓN DEL CURSO: 2-L-1
# PROFESOR DE TEORÍA: FELIPE MORENO
# PROFESOR DE LABORATORIO: JUAN PADILLA
#
# AUTOR
# NOMBRE: Juan Carlos Perez Gonzalez
# RUT: 20.345.432-2
# CARRERA: Ingeniería Civil Mecánica
# DESCRIPCIÓN DEL PROGRAMA ...<CONTINÚE CON EL PROGRAMA A PARTIR DE AQUÍ>
```

Código 1: Ejemplo de encabezado

PROBLEMA

El Servicio Electoral de Chile (SERVEL) es la institución encargada de la administración, supervisión y fiscalización de los procesos electorales y plebiscitarios del país. Es un organismo autónomo y una de las instituciones que está mejor valorada entre la ciudadanía, por la transparencia de los procesos electorales y la velocidad en la que se entregan los resultados.

Sin embargo, los procesos electorales en el país no están exentos de polémica y problemas de gestión, es por ello, que se le ha solicitado a usted, futuro ingeniero de la Universidad de Santiago de Chile, que apoye a la resolución de los problemas que existen actualmente en la revisión y cálculo de resultados de los procesos electorales.

Cuando ocurre un proceso electoral, los locales de votación van enviando por un sistema interno los resultados de las mesas que ya han contabilizado los votos, lo que genera en la base de datos un archivo llamado “mesa-x.txt”, donde x corresponde a un número que se asigna al archivo al momento de ser ingresado, es decir, la primera mesa en reportar sus

resultados entra al sistema entra como mesa-1.txt y luego, las que van llegando toman los números siguientes. Los archivos contienen:

- En la primera fila y separados por coma: la comuna a la que pertenece la mesa, el número de mesa (que corresponde a un identificador de un número, que puede ser de varios dígitos, más una letra, históricamente V para varones y M para mujeres), la cantidad de sufragios emitidos y el total de personas habilitadas para sufragar registradas en la mesa
- Luego, cada fila del archivo corresponde a un candidato (en el orden en el que aparecen en la papeleta) y la cantidad de votos contabilizados para él. Separados por coma.
- Finalmente, las dos últimas filas contienen la cantidad de votos nulos y blancos en la mesa.

Un ejemplo de archivo de este tipo puede revisarse a continuación (Figura 1).

mesa-48.txt
LO PRADO,162V,51,100
JUAN PÉREZ GONZÁLEZ,24
AMANDA ROJO PEÑA,20
PEDRO RAMIREZ CALDERÓN,3
JUAN MANUEL ZUÑIGA LORCA,3
NULOS,1
BLANCOS,0

Figura 1 - Ejemplo de archivo de entrada

El problema es que, dada la forma en que llegan los archivos, resulta difícil entregar resultados parciales de la votación por localidad, pues hay que buscar entre todos los archivos, los de cada comuna en particular para poder realizar el cálculo.

Por otro lado, la forma en que se ingresan las mesas provoca que la revisión de mesas descuadradas, aquellas en dónde la suma de votos totales no cuadra con el total de votos emitidos, sea lenta y poco confiable.

A fin de solucionar ambos problemas, SERVEL le ha pedido a usted que construya un programa capaz de generar automáticamente un informe, en formato .txt, para cada comuna que haya reportado a lo menos una mesa, el cuál contenga:

- Información general: Informando la cantidad de mesas escrutadas, la cantidad de mesas correctamente cuadradas y la cantidad de mesas descuadradas.
- Resultados parciales: con el cálculo de votos totales y porcentuales para cada candidato, sin considerar para efectos de cálculo las mesas descuadradas.
- El listado de mesas con descuadre, para informar a los locales de votación y revisar nuevamente los conteos, indicando el número de mesa y **no** el nombre del archivo en el sistema.

Se le ha provisto de un ejemplo de este archivo en la figura 2.

Para facilitar su solución considere que el usuario, al momento de ejecutar el programa, indica como entrada el número del último archivo a revisar. Por ejemplo, si quisiera revisar

un conjunto de 50 archivos, el usuario ingresa el valor 50 para que el programa genere el reporte utilizando todos los archivos de mesa desde el mesa-1.txt hasta el mesa-50.txt.

LO PRADO.txt
GENERAL ##### MESAS ESCRUTADAS: 26 CORRECTAS: 23 DESCUADRADAS: 3 ##### RESULTADOS PARCIALES ##### JUAN PÉREZ GONZÁLEZ,600,47.66% AMANDA ROJO PEÑA,502,39.87% PEDRO RAMIREZ CALDERÓN,64,5.08% JUAN MANUEL ZUÑIGA LORCA,61,4.85% NULOS,12,0.95% BLANCOS,20,1.59% ##### MESAS DESCUADRADAS ##### 12M 45V 65M

Figura 2 - Ejemplo de archivo de salida

Considere para su implementación que:

- Los resultados porcentuales deben entregarse con dos decimales y el signo de porcentaje.
- Si bien puede haber candidatos con 0 votos, o que no haya votos nulos o blancos, el archivo de cada mesa contemplará siempre el listado completo de cada candidato y las líneas de nulos y blancos indicando, para esos casos, 0.
- El programa debiera funcionar para una cantidad indeterminada de archivos, e independientemente del número de candidatos para cada elección.
- Para facilitar su solución, considere que cada conjunto de archivos entregado tiene siempre los mismos candidatos, es decir:
 - Si la elección fuera de alcalde o concejales, todos los archivos serían de la misma comuna.
 - Si la elección fuera de diputados, todos los archivos serían de comunas del mismo distrito.
 - Si la elección fuera presidencial, podría haber archivos de cada comuna del país.

Restricciones de implementación

Para esta prueba se prohíbe el uso de:

- Funciones y métodos para obtener mínimos y máximos, así como para ordenar listas.
- Construcciones que usen las palabras reservadas `try` y `except`.
- Expresiones regulares.
- Listas por comprensión.
- Uso de strings con formato para imprimir, ya sea utilizando la sintaxis de llaves `{, }` o de porcentaje `%`.
- Los tipos de dato conjunto (`set`), tupla (`tuple`) y diccionario (`dict`).
- Módulos que no han sido vistos en clases de teoría (e.j. `pandas`, `csv`, etc).
- Programación orientada a objetos, es decir, definición de clases y métodos.
- Las funciones `map()` y `filter()`.
- El uso de funciones del módulo `OS`.

OBSERVACIONES

En caso de que su equipo le entregue el siguiente error al leer o escribir el archivo:

UnicodeDecodeError: 'charmap' codec can't decode byte

Utilice el parámetro `encoding='utf-8'` luego del modo de apertura al utilizar la función `open()`.

EVALUACIÓN

Considere que la evaluación considera 60 puntos como puntaje máximo y una exigencia del 60%.

Los aspectos a evaluar de la solución son:

- Abstracción de datos.
- Abstracción de procesos.
- Buenas prácticas de programación.
- Organización del código.
- Manejo de estructuras de control.
- Manejo de tipos de datos.
- Uso de funciones.
- Lectura y escritura de archivos.
- Flujo de programa.
- Cálculo de la solución.