

FACULTAD DE INGENIERÍA
FUNDAMENTOS DE COMPUTACIÓN Y PROGRAMACIÓN
PRUEBA PARCIAL PROGRAMADA N°2 – SERIE 1



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 calificados 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.
- Estudiantes que hayan sido sancionados con copia en evaluaciones anteriores y vuelvan a incurrir en actos similares en esta evaluación serán sancionados con la reprobación directa de la asignatura.
- 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 28 de diciembre de 2020, 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 formato IDÉNTICO 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

Contexto

La pandemia del Covid-19 ha afectado severamente todas las labores a nivel mundial, principalmente a raíz de las medidas de confinamiento establecidas por los diferentes países.

Si bien el Viejo Pascuero ha sido declarado personal esencial en diversos países, este año la Navidad estuvo en serios problemas puesto que no se contaba con un plan adecuado para el reparto de regalos. Puesto que se prevé que la pandemia persista para el año 2021, el equipo asesor del Viejo Pascuero ha decidido automatizar el proceso de asignación de regalos a cada niño a fin de destinar a los duendes encargados de esta labor a la elaboración de un plan más adecuado para el reparto de regalos que considere las medidas dispuestas en cada territorio.

Problema

A raíz de su buen comportamiento durante el año 2020, el Viejo Pascuero ha decidido encomendarle a usted el desarrollo del software de gestión de regalos, para lo cual le hace entrega de los siguientes elementos:

- Archivo “bondad.txt”.
- Archivo “pedidos.txt”.

Adicionalmente, como regalo, le hace entrega del módulo Jojojo.py, el cual contiene algunas funciones que necesitará, sabiendo que están más allá de sus conocimientos actuales.

El archivo “bondad.txt”, elaborado por el Departamento de Calificación del Nivel de Bondad, contiene en cada línea el nombre completo de un niño y una nota entre 1.0 y 7.0 por cada mes del año. Las notas están separadas por una coma.

Ejemplo:

bondad.txt	
Aquiles Baeza,	5.1,2.0,4.8,7.0,6.1,4.7,6.1,5.4,3.8,3.7,6.6,6.1
Esteban Quito,	6.1,6.6,6.3,7.0,6.0,6.8,6.7,5.5,5.4,6.3,6.5,6.8
Elsa Payo,	3.0,2.1,3.9,4.5,3.6,4.2,3.7,4.8,1.3,1.5,2.4,3.3
Armando Casas,	5.8,6.1,7.0,5.2,4.0,3.6,5.1,6.9,5.4,4.0,3.6,1.2

El archivo “pedidos.txt”, creado por el Departamento de Cartas y Pedidos, contiene en cada línea el nombre completo de un niño y la lista de regalos que pidió. Los elementos del archivo están separados por una coma.

Ejemplo:

pedidos.txt	
Aquiles Baeza,	bicicleta, pelota de futbol, celular, xbox, camion, televisor
Esteban Quito,	transformer, lego
Elsa Payo,	monster high, my little pony, hello kitty, Barbie
Zacarias Labarca del Rio,	nintendo switch

Se le solicita crear el listado de regalos a repartir, el cual debe ser almacenado en el archivo “regalos.txt”, que en cada línea contenga el nombre completo del niño y los regalos que recibirá, con los elementos separados por una coma.

Ejemplo:

regalos.txt
Aquiles Baeza,bicicleta
Esteban Quito,transformer,lego
Elsa Payo,carbon
Zacarias Labarca del Rio,nintendo switch

La cantidad máxima de regalos que recibe cada niño se determina de acuerdo a la siguiente tabla, dependiendo de la bondad anual promedio de cada niño:

Bondad promedio	Regalos
6.5 – 7.0	5
6.0 – 6.4	4
5.5 – 5.9	3
5.0 – 5.4	2
4.0 – 4.9	1
1.0 – 3.9	Carbón

Note que aquellos niños que se portaron mal no reciben regalos, sino carbón. A su vez, los regalos que el niño recibe serán tomados de la lista de pedidos en orden de aparición. Tenga en cuenta que:

- Si el niño pidió menos regalos de los que podría recibir, solo recibe lo que pidió.
- Si un niño no pidió regalos, pero debe recibir a lo menos uno porque se portó bien, se le asigna como regalo una bicicleta.
- Si alguien pidió regalos, pero no figura en el archivo “bondad.txt”, no debe ser considerado.

Implemente el programa que ayuda al Viejo Pascuero en su asignación de regalos.

Indicaciones adicionales

Para poder usar el módulo Jojojo, copie el archivo Jojojo.py en la misma carpeta en que se encuentre el código fuente de su programa. Este módulo cuenta con dos funciones:

- *leer_archivo(nombre_archivo)*: recibe como entrada un string con el nombre del archivo a leer, incluyendo la extensión. Dicho archivo debe estar en la misma carpeta que el código fuente de su programa. Entrega como salida una lista de strings con cada línea del archivo, incluyendo el salto de línea final.

- `escribir_archivo(nombre_archivo, lineas)`: recibe como entradas el nombre del archivo a leer (incluyendo la extensión) y una lista con las líneas de texto a escribir en el archivo. Cada línea debe contener el salto de línea final. No entrega una salida, pero crea el archivo de texto con el contenido en la misma carpeta que el código fuente de su programa.

No copie las funciones en su código, pues como no son de su autoría, debe importarlas desde el módulo entregado. Del mismo modo, no puede modificar el contenido del módulo “Jojojo.py” pues su solución será probada con las funciones del módulo como le fue originalmente entregado.

Para el bloque principal, considere como entrada la lectura de archivos y como salida, la escritura del archivo solicitado.

Tenga en cuenta que el uso de funciones en la solución es **OBLIGATORIO**.

La solución debe funcionar para cualesquiera archivos de entrada que respeten el formato establecido, y no solo para los que se proporcionan como ejemplo, independiente de la cantidad de registros (filas) que estos contengan.

Restricciones de implementación

Para esta prueba, se prohíbe el uso de las siguientes herramientas de programación:

- Funciones y métodos para ordenar o para obtener máximos y mínimos.
- Las funciones nativas `map()` y `filter()`.
- Tipos de dato conjunto y diccionario.
- Ciclos `for` y listas por comprensión.
- Uso de strings con formato, ya sea utilizando la sintaxis de llaves `{, }` o de porcentaje `%`.
- Construcciones para manejo de excepciones (`try, except`).
- Definición de sus propias funciones/métodos para leer o escribir archivos.
- Programación orientada a objetos, es decir, definición de clases y métodos.

El uso de alguna de estas herramientas significará la nota mínima.

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.
- Flujo de programa.
- Cálculo de la solución.