

Trabajo Práctico Grupal Obligatorio – Segundo cuatrimestre 2019

Fecha límite de 1era. entrega: **22 / 09 / 2019**

Fecha límite de 2da. entrega: **26 / 10 / 2019**

Fecha límite de 3ra. entrega: **18 / 11 / 2019**

Fecha límite de entrega final: **01 / 12 / 2019**

Problema a resolver

La empresa desarrolladora de software “FoolMatic” desea presentarse a una licitación, para la provisión de un sistema de recuento de votos, con carga manual de totales por mesa, para una escuela de la C.A.B.A. En este contexto, ha solicitado a los estudiantes de Informática de UNDAV el desarrollo de un prototipo de software que permita, al usuario responsable del sistema, ingresar datos de telegramas de votos emitidos en la escuela, de acuerdo a las siguientes particularidades.

Además de la fecha y datos de la escuela (número, nombre, sección y circuito), se ingresa, por cada telegrama, el número de mesa electoral y para cada agrupación o clase de voto (Alfa, Beta, En Blanco), la cantidad de votos para la categoría “Jefe de Gobierno” obtenidos en la mesa, por esa agrupación o clase de voto.

Se requiere un sistema programado en Python que ofrezca las siguientes prestaciones:

1. Solicite el ingreso del número de sección electoral, el número de circuito, el nombre y número de escuela y la fecha del acto electoral (año, mes y día); por otra parte, también debe solicitar las cantidades de votos de cada agrupación (o clase de voto) en cada mesa.
2. Asegure la validez de día, mes y año (de 2015 a 2027 incluidos), de sección electoral (entre 1 y 15), de circuito (entre 1 y 167), de mesa (entre 1 y 7413) y que la cantidad de votos para cada agrupación no supere el valor 350 en cada mesa. Cuando un dato sea incorrecto, deberá reingresarse hasta que resulte válido, para poder continuar.
3. Muestre simultáneamente en pantalla todos los datos validados de cada telegrama (mesa, agrupación o clase de voto, categoría, cantidad de votos), a medida que son ingresados.
4. Determine y muestre, junto a datos de la escuela, la cantidad total de votos.
5. Determine y muestre la cantidad de votos obtenidos por cada agrupación y su respectivo porcentaje sobre el total (excluyendo del mismo, la cantidad de votos en blanco), como así también, la cantidad total de votos en blanco.

Es altamente probable que, en una etapa posterior, se requiera modificar el sistema para que genere más prestaciones (nuevas estadísticas, restricciones, etc.).

Documentar, diseñar e implementar en Python, un programa (con todas sus funciones) que satisfaga el requerimiento citado anteriormente, respetando las condiciones de presentación que se describen a continuación.

Condiciones de presentación

Antes de concluir el plazo preestablecido por cada fecha límite, se debe presentar el trabajo en 4 entregas, con las características que se describen más abajo. Cada entrega constará de **un documento en procesador de textos** en el que se informen pasos de la metodología y, además, en la 3ra. y 4ta. entrega, **cuatro** (o más) **archivos en formato .py** (código Python). El **documento en procesador de textos** debe contener:

- a) Una **carátula** con los siguientes datos:
 - Carrera y Nombre de la materia
 - Año, Cuatrimestre, Nro. de Comisión y Docentes
 - Apellido y nombres de cada estudiante (hasta 5 integrantes por grupo y no menos de 4)
- b) **Informe** con el análisis del problema y los pasos de la solución requeridos en las sucesivas entregas. Dichos pasos están enunciados en la “metodología para la construcción de programas pequeños”, que puede leerse en el ítem 2.1 de la Unidad 2 del apunte “Introducción a Python - Parte I”.

✓ **1era. Entrega** Fecha límite de presentación: **domingo 22 de septiembre**

A. **Analizar el problema.** Entender profundamente cuál es el problema que se trata de resolver, incluyendo el contexto en el cual se usará. Una vez analizado el problema, asentar el análisis por escrito.

B. **Especificar la solución.** Describir qué debe hacer el programa (incluyendo validaciones), sin importar el cómo. Determinar cuáles son los datos de entrada que se proveen, cuáles son las salidas que se deben producir. Documentar dicha especificación por escrito.

✓ **2da. Entrega** Fecha límite de presentación: **sábado 26 de octubre**

A y B. **Análisis del problema y Especificación de la solución** (corregidos y aprobados)

C. **Diseñar la solución.** Describir cómo resolver el problema, cuáles son los algoritmos y las estructuras de datos que se usarán, cuáles son las funciones en que se divide el problema y cuál es la relación entre ellas y/o con el programa (datos de entrada y salidas respectivas). Analizar posibles variantes y tomar las decisiones usando como dato de la realidad el contexto en el que se aplicará la solución y los costos asociados a cada diseño. Asentar por escrito el diseño, asegurándose de que esté completo, incluyendo la especificación de cada función.

✓ **3ra. Entrega** Fecha límite de presentación: **lunes 18 de noviembre**

A, B y C. **Análisis del problema, Especificación y Diseño de la solución** (corregidos y aprobados)

D. **Implementar el diseño.** Traducir a un lenguaje de programación (en nuestro caso Python) el diseño que fue elegido en el punto anterior. Enviar por mail en formato .py el código Python (un módulo con el programa principal y, por cada integrante del grupo, el respectivo módulo con funciones).

E. **Probar el programa.** Diseñar un conjunto de pruebas que verifique el comportamiento de cada una de sus partes por separado, y también la correcta integración entre ellas. Utilizar la instrucción print “DEBUG” como instrumento de depuración para descubrir eventuales errores. Documentar casos que garanticen que todas las componentes estén probadas con datos “correctos”, “incorrectos” y “de borde”.

✓ **4ta. Entrega** Fecha límite de presentación: **domingo 1 de diciembre**

F. **Mantener el programa.** Adaptar el programa para dar respuesta al nuevo requerimiento. Especificar, implementar y probar las nuevas funcionalidades del programa. Informe completo (A, B, C, D, E y F)

Todos los archivos (informe en procesador de textos y archivos de código Python) deben ser **enviados por uno de los integrantes** del equipo, mediante correo electrónico, a los docentes de la Comisión, con copia a jsabellarosa@undav.edu.ar y a los demás integrantes del propio grupo, enunciando como asunto del correo “**UNDAV-TP Informática-Comisión ...**” y los apellidos de los miembros del grupo.