

## **Programación III**

### **Proyecto Segundo parcial**

#### **Introducción:**

Un cliente de correo es un software que permite realizar la gestión de los correos recibidos y enviados. Nos permite buscar, ordenar y filtrar mails según los criterios que nos sean mas convenientes.

#### **Enunciado:**

Se desea realizar una aplicación que facilite el trabajo de gestión de los e-mails de la bandeja de entrada. La aplicación debe permitir la búsqueda y visualización de los mails de diferentes maneras. Se podrá agregar y quitar mails al sistema debiendo ser guardados en la/las estructura/as de dato/os mas conveniente/es.

El programa deberá tener las siguientes funcionalidades:

- Agregar mails al gestor
- Borrar mails del gestor
- Mostrar mails ordenados por fecha
  - Permitir filtrar mails por rango de fechas
- Mostrar mails ordenados por remitente
- Buscar mails por remitente
- Buscar mails por palabras del texto o asunto

El docente le proveerá una plantilla del proyecto a los alumnos con la clase "MailManager". Los alumnos deberán implementar la funcionalidad de la clase y utilizar las mejores estructuras y algoritmos para resolver el problema.

El alumno debe implementar un programa Principal con la interfaz de usuario de la aplicación, que permita usar todas las funcionalidades de la clase MailManager. Queda a criterio del alumno implementarla de modo gráfico o modo texto.

La evaluación se realizará sobre el correcto funcionamiento, implementación y rendimiento de la clase MailManager. El entorno de usuario debe estar presente pero no se evaluará su implementación o formato.

Todas las estructuras y algoritmos utilizados en la clase MailManager deben ser implementados por el usuario, no está permitido utilizar bibliotecas STL, como List, Map, Queue, SortedMap, Sort, etc.

#### **Objetivo:**

Ejercitar y afianzar el uso de estructuras de datos mediante una aplicación concisa y de uso general. Afianzar la importancia del uso de las estructuras correctas para mejorar el rendimiento de las aplicaciones.

**Formas de presentación:** Exposición y defensa del programa e informe con código fuente y comentarios.

**Forma de trabajo:** Grupos de no más de 2 personas.

**Forma de evaluación:** Se evaluará de acuerdo a los siguientes parámetros:

- Correcto funcionamiento del programa
- Completitud de las funcionalidades del enunciado
- Presentación (interfaz, código, prolijidad)
- Solución óptima al problema dado – eficiencia del código.
- Utilización y adaptación de los códigos proporcionados en clase y del libro.
- Conocimiento de la estructura utilizada.
- Exposición y defensa del trabajo.
- Preguntas sobre las estructuras utilizadas a cada integrante.

**Cada grupo tendrá 15 minutos para presentar lo desarrollado el día de la entrega.**

**La plantilla y creación de grupo esta disponible en la siguiente URL:**

<https://classroom.github.com/g/JcBOzoGE>