

## **Trabajo Práctico Especial** **Programación Imperativa** **Diciembre 2018**

### **1. Objetivo**

Diseñar e implementar, en grupos de hasta tres alumnos, un programa para el procesamiento de datos de vuelos desde y hacia aeropuertos nacionales, basado en datos reales. Para ello se deberá realizar tanto el front-end como el back-end, este último basado en la creación de al menos un TAD.

Los datos que alimentarán nuestro dataset son datos oficiales elaborados por la EANA y almacenados en archivos CSV/DSV ( ver [https://en.wikipedia.org/wiki/Comma-separated\\_values](https://en.wikipedia.org/wiki/Comma-separated_values) ). La información a procesar proviene de datos oficiales del Ministerio de Transporte: <http://www.datos.gob.ar/en/dataset/transporte-aterrizajes-despegues-registrados-por-eana>

### **2. Descripción funcional**

El programa deberá leer de dos archivos csv los datos de los aeropuertos nacionales y de aterrizajes y despegues registrados por el EANA, donde el delimitador de campos es un punto y coma. Los archivos tienen las siguientes características:

Detalle de aeródromos y helipuertos, a partir de ahora **aeropuertos.csv**

❖ **Origen:**

[https://servicios.transporte.gob.ar/gobierno\\_abierto/detalle.php?t=aeropuertos&d=detalle](https://servicios.transporte.gob.ar/gobierno_abierto/detalle.php?t=aeropuertos&d=detalle)

❖ **Descarga: En Campus ITBA**

❖ **Campos Relevantes:**

- **oaci:** El código de aeropuertos de OACI es el código de designación de aeropuertos compuesto de cuatro caracteres alfanuméricos que sirve para identificar cada aeropuerto alrededor del mundo. Si está en blanco indica que el aeropuerto no tiene código OACI. [https://es.wikipedia.org/wiki/Código\\_de\\_aeropuertos\\_de\\_OACI](https://es.wikipedia.org/wiki/Código_de_aeropuertos_de_OACI)
- **denominacion:** nombre del aeropuerto. Máximo de 70 caracteres. Puede contener letras, símbolos, acentos, ñ, espacios
- **provincia:** nombre de la provincia donde se ubica el aeropuerto.

El archivo se compone de una primera línea de encabezado, con los títulos de cada campo. De la segunda línea en adelante, cada línea representa un aeropuerto (o similar) conteniendo los datos de cada una de los campos, separados por ;.

Ejemplo de primeras 3 líneas de archivo, con títulos e información de los aeropuertos de "AZUL" y "BELL VILLE/LA ZULEMA".

```
local;oaci;iata;tipo;denominacion;coordenadas;latitud;longitud;elev;uom_elev;ref;distancia_ref;direccion_ref;condicion;control;region;fir;uso;trafico;sna;concesionado;provincia;inhab
ZUL;SAZA;;Aeródromo;AZUL;"36°50'12""S
59°52'51""W";-59.88080000;-36.83675000;146.00;Metros;Azul;3.0;S0;PUBLICO;
NOCONTROL;RACE;SAEF;CIVIL;Nacional;NO;NO;"BUENOS AIRES";NO
ZLM;;;Aeródromo;"BELL VILLE/LA ZULEMA";"32°29'23""S
62°40'17""W";-62.67138889;-32.48972222;129.00;Metros;"Bell
Ville";13.0;N;PRIVADO;NOCONTROL;RANO;SACF;AEROAPP;Nacional;NO;NO;CÓRDOBA;
NO
```

A partir de ahora denominaremos aeropuerto a cualquier establecimiento del archivo aeropuertos.csv sin considerar el campo Tipo (es decir, sin importar si es Aeródromo o Helipuerto).

Detalle de movimientos de aterrizajes y despegues, a partir de ahora **movimientos.csv**

- ❖ **Origen:** [https://servicios.transporte.gob.ar/gobierno\\_abierto/detalle.php?t=eana&d=detalle](https://servicios.transporte.gob.ar/gobierno_abierto/detalle.php?t=eana&d=detalle)
- ❖ **Descarga:** En Campus ITBA
- ❖ **Campos Relevantes:**
  - **Fecha:** la fecha del movimiento, en forma "dd/mm/yyyy"
  - **Clase de vuelo:** Regular, No Regular, Vuelo Privado con Matrícula Extranjera y Vuelo Privado con Matrícula Nacional.
  - **Clasificación Vuelo:** Cabotaje, Internacional y N/A.
  - **Tipo de Movimiento:** Despegue y Aterrizaje.
  - **Origen OACI:** Código OACI del aeropuerto del cual despegó el vuelo. Dato obligatorio, no está en blanco.
  - **Destino OACI:** Código OACI del aeropuerto en el cual aterrizó el vuelo. Tanto este como el anterior pueden ser aeropuertos de otros países y por lo tanto no figurar en el archivo de aeropuertos. Es un dato obligatorio, no está en blanco. Tanto para este campo como el anterior, si el código tiene la forma "AR-#####" o "SA###" donde # representa un dígito, entonces no corresponde a un aeropuerto conocido, es decir, no estará presente en aeropuertos.csv.
  - **Aerolínea Nombre:** El nombre de la aerolínea. Puede ser N/A si no pertenece a una línea aérea, o estar en blanco si no es un vuelo regular.

El archivo se compone de una primera línea de encabezado, con los títulos de cada campo. De la segunda línea en adelante, cada línea representa un movimiento (aterrizaje o despegue) conteniendo los datos de cada una de los campos, separados por ;.

Ejemplo de las primeras 4 líneas del archivo:

```
Fecha;Hora Movimiento HHMM;Clase de Vuelo;Clasificación Vuelo;Tipo de
Movimiento;Origen OACI;Destino OACI;Aerolínea Nombre;Aeronave;Aeronave
APC
01/01/2014;00:01;Regular;Internacional;Aterrizaje;LEMD;SAEZ;Iberia
Airlines;AIRBUS A-340;D
01/01/2014;00:07;Regular;Internacional;Aterrizaje;SBGR;SAEZ;LATAM
Argentina;AIRBUS A-320;C
01/01/2014;00:18;Regular;Cabotaje;Aterrizaje;SAWG;SABE;Austral Líneas
Aéreas;EMBRAER E-190;C
```

En la segunda línea, un vuelo internacional de Iberia Airlines que aterrizó a las 00:01 del 1 de enero de 2014 en el aeropuerto de Ezeiza procedente del aeropuerto cuyo código OACI es LEMD. El código LEMD identifica al aeropuerto de Barajas, Madrid. El aeropuerto LEMD no estará presente en el archivo aeropuertos.csv ya que el mismo contiene únicamente aeropuertos nacionales.

Se considera “Movimiento” al despegue o aterrizaje de un avión en un aeropuerto. Para los propósitos de tráfico del aeropuerto, una llegada y una salida se cuentan como dos movimientos.

*Como los datos provienen de un sitio oficial, se asume que el formato y contenido de los archivos es correcto.*

**El programa deberá recibir por línea de comando (no por entrada estándar) el path del archivo de aeropuertos y luego el path del archivo de movimientos.**

Por ejemplo, si ambos archivos CSV se llaman `aeropuertos.csv` y `movimientos.csv` y están en el mismo directorio que el ejecutable, el programa se puede invocar cómo:

```
$> ./censo aeropuertos.csv movimientos.csv
```

Si ambos archivos CSV se llaman `aerop.csv` y `movs.csv` y están en el directorio superior al ejecutable, se invocará como

```
$> ./censo ../aerop.csv ../movs.csv
```

**El programa deberá resolver las siguientes consultas (*queries*) dejando los resultados de cada una en archivos .csv localizados en el mismo directorio que el del ejecutable.**

### **Query 1: Movimientos por aeropuerto**

Donde cada línea de la salida contenga separados por ; el código OACI, la denominación (obtenida del archivo `aeropuertos.csv`) y la cantidad de movimientos del aeropuerto (contabilizando únicamente despegues que tengan al aeropuerto como origen y aterrizajes que tengan al aeropuerto como destino).

El orden de impresión es descendente por movimientos y luego alfabéticamente por código OACI.

Sólo se deberán listar los aeropuertos presentes en `aeropuertos.csv`.

- ☐ Nombre del archivo: `movimientos_aeropuerto.csv`
- ☐ Salida de ejemplo:

```
OACI;Denominación;Movimientos
SABE;BUENOS AIRES/AEROPARQUE J. NEWBERY;10289
SAEZ;EZEIZA/MINISTRO PISTARINI;6039
SADF;SAN FERNANDO;2860
...
```

### **Query 2: Movimientos por día de la semana, ordenado cronológicamente por el día de la semana**

Donde cada línea de la salida contenga separados por ',' el día de la semana (obtenido a partir del campo Fecha), la cantidad de movimientos de cabotaje para ese día, la cantidad de movimientos internacionales para ese día y la suma de ambos movimientos.

- ☐ Nombre del archivo: `dia_semana.csv`
- ☐ Salida de ejemplo:

```
Día;Cabotaje;Internacional;Total
```

```

lunes;523;291;814
martes;425;222;647
miércoles;525;301;826
jueves;543;307;850
viernes;558;300;858
sábado;442;287;729
domingo;465;301;766

```

- ❑ Sugerencia: Apéndice B10 Funciones de fecha y hora <time.h> en Kernighan, B. & Ritchie, D., El Lenguaje de Programación C, Prentice Hall

### **Query 3: Composición de Movimientos**

Donde cada línea de la salida contenga separados por ‘;’ el total de **movimientos** por Clasificación de Vuelo (con las categorías Cabotaje e Internacional) y por Clase de Vuelo (con las categorías Regular, No Regular y Vuelo Privado).

La clase de vuelo “Vuelo Privado” contempla a las clases de vuelo “Vuelo Privado con Matrícula Nacional” y “Vuelo Privado con Matrícula Extranjera”.

- ❑ Nombre del archivo: **composicion.csv**
- ❑ Salida de ejemplo:

```

Clasificación de Vuelo;Clase de Vuelo;Movimientos
Cabotaje;Regular;100
Cabotaje;No Regular;10
Cabotaje;Vuelo Privado;1
Internacional;Regular;50
Internacional;No Regular;5
Internacional;Vuelo Privado;1

```

***Para los grupos que entreguen en segunda fecha, además de las estadísticas arriba mencionadas, deben implementar las siguientes:***

### **Query 4: Pares de provincias que comparten al menos 1000 movimientos**

Donde cada línea de la salida contenga separados por ; el par de provincias distintas (obteniendo los nombres de las mismas del archivo `aeropuertos.csv`) y la cantidad de movimientos de todo tipo que comparten ambas provincias.

Dos provincias A y B comparten un movimiento si la provincia del aeropuerto de origen/destino OACI del movimiento es A y la provincia del aeropuerto de destino/origen OACI del movimiento es B.

El orden de impresión es descendente por movimientos y el orden de los pares es alfabético por nombre.

Se deben listar únicamente los pares de provincias distintas que comparten 1000 o más movimientos.

No se deben listar los pares opuestos. Es decir, si se lista el par Provincia A;Provincia B;Movimientos no se debe listar el par Provincia B;Provincia A;Movimientos.

La implementación no debe estar limitada a las provincias argentinas actuales (dependerá de las provincias mencionadas en el archivo `aeropuertos.csv`).

- ❑ Nombre del archivo: pares\_provincias.csv
- ❑ Salida de ejemplo:

```
Provincia A;Provincia B;Movimientos
CIUDAD AUTÓNOMA DE BUENOS AIRES;CÓRDOBA;83542
BUENOS AIRES;CIUDAD AUTÓNOMA DE BUENOS AIRES;73430
CIUDAD AUTÓNOMA DE BUENOS AIRES;MENDOZA;71336
CIUDAD AUTÓNOMA DE BUENOS AIRES;SANTA FÉ;33211
BUENOS AIRES;CÓRDOBA;30325
BUENOS AIRES;SANTA FÉ;29861
...
```

### **Query 5: Aerolíneas según el porcentaje de movimientos de cabotaje, ordenado descendentemente por el porcentaje de movimientos de cabotaje**

Donde cada línea de la salida contenga separados por ‘;’ el nombre de la aerolínea y el porcentaje de movimientos donde Clasificación de Vuelo es Cabotaje de esa aerolínea respecto del total de movimientos de cabotaje de todas las aerolíneas.

Descartar aquellos movimientos de cabotaje que no tengan asociado un Aerolínea Nombre o el mismo sea N/A.

- ❑ Nombre del archivo: porcentaje\_cabotaje.csv
- ❑ Salida de ejemplo:

```
Aerolínea;Porcentaje
Aerolíneas Argentinas;66%
Austral Líneas Aéreas;14%
LATAM Argentina;12%
SOL Líneas Aéreas;4%
Andes Líneas Aéreas;4%
```

*Para todas las queries, tener en cuenta que todos los archivos de salida deben contener la línea de encabezado correspondiente indicada en las salidas de ejemplo.*

## **3. Contenidos**

Para la realización del presente trabajo se recomienda consultar el siguiente material:

- Para leer argumentos por línea de comandos:  
[https://www.tutorialspoint.com/cprogramming/c\\_command\\_line\\_arguments.htm](https://www.tutorialspoint.com/cprogramming/c_command_line_arguments.htm)
- strtok puede resultar útil para "parsear" las líneas de los archivos:  
[https://www.tutorialspoint.com/c\\_standard\\_library/c\\_function\\_strtok.htm](https://www.tutorialspoint.com/c_standard_library/c_function_strtok.htm)
- **Presentación de Archivos:** Contiene lo esencial para el manejo de archivos en C. Tener en cuenta que se procesarán únicamente archivos de texto en forma secuencial, por lo que las funciones a utilizar serán fopen, fgets, fclose, fprintf o similares. Se encuentra en Campus ITBA.

De todas formas los alumnos pueden consultar dudas relacionadas a estos nuevos contenidos usando los foros de discusión Campus ITBA. No se responderán consultas que no sean formuladas a través del Campus (excepto para los alumnos que hayan cursado en cuatrimestres anteriores)

#### **4. Diseño e Implementación del Programa**

Se debe realizar un diseño donde se separe claramente la lectura y preparación de datos (front-end) del almacenamiento y procesamiento de los datos (back-end).

Para el almacenamiento de los datos se deberá desarrollar al menos un TAD.

En ningún caso se debe repetir código para resolver situaciones similares, sino que debe implementarse una correcta modularización y se deben reutilizar funciones parametrizadas.

Tanto la biblioteca como el front-end deben estar correctamente comentados y en el caso de la biblioteca se debe escribir el archivo de encabezado correspondiente.

El programa no debe presentar leaks de memoria.

El programa no debe abortar por ningún motivo y ante cualquier error se debe mostrar un mensaje adecuado.

Se tendrá en cuenta la eficiencia en el uso de recursos, tanto el tiempo de ejecución como la memoria utilizada.

#### **5. Material a entregar**

Cada grupo deberá subir al Campus ITBA un archivo compactado conteniendo como mínimo los siguientes archivos:

- Archivos fuentes y de encabezado
- Archivo de texto README con la explicación de cómo generar el ejecutable y de como ejecutar el programa
- makefile. Consultar apunte en Campus ITBA.

#### **6. Armado de Grupos**

Los alumnos deben informar la conformación del grupo y la fecha de final elegida en el Foro de Discusión “**TPE Final**” de Campus ITBA antes del **05/12/2018 12:00 ART**.

Cada grupo deberá realizar la entrega mediante la actividad correspondiente en Campus ITBA:

- **En caso de rendir en la primera fecha de final (10/12/2018 08:00 ART) deberán entregar antes del 07/12/2018 23:59 ART**
- **En caso de rendir en la segunda fecha de final (18/12/2018 08:00 ART) deberán entregar antes del 15/12/2018 23:59 ART**

En ambos casos enviar sólo el material pedido. No incluir código compilado, archivos de prueba publicados, etc.

En ambos casos el grupo completo deberá presentarse el día y hora del final donde se le comunicará el resultado del mismo y rendir un coloquio, el cual podrá modificar la nota individual de los integrantes del grupo.

Para que el trabajo sea aceptado todos los integrantes del grupo deben estar inscriptos en fecha de final que acordaron en el armado del grupo.

En caso de inscribirse y no entregar en fecha, se calificará como ausente. Si algún miembro del equipo no se presenta en el horario del final fijado por Secretaría Académica el alumno será

calificado como ausente, y deberá conformar otro grupo en otra fecha de final, no afectando la evaluación de los alumnos que se presenten.

## 7. Criterios de Evaluación y Calificación

El programa debe poder compilarse y ejecutarse en Pampero usando **gcc** con los parámetros **-pedantic -std=c99 -Wall -fsanitize=address**. El último flag verifica que la memoria no sea usada en forma incorrecta (extenderse de la cantidad de bytes reservados en una zona, acceder a una zona de memoria liberada, no liberar memoria antes de finalizar la ejecución, etc.)

No se aceptará el uso de bibliotecas de terceros, a excepción de la biblioteca estándar de C. El código debe ser íntegramente de autoría propia. El uso de bibliotecas no autorizadas implicará la desaprobación.

Si un trabajo presenta errores de compilación, el mismo será reprobado. Se espera que no se presenten “*warnings*” evitables.

### **Para la evaluación y calificación del trabajo especial se considerarán:**

- **el cumplimiento del modo de ejecución y de los nombres de archivos pedidos**
- **el correcto funcionamiento del programa**
- **la modularización realizada**
- **el correcto diseño del TAD, que debería poder ser fácilmente adaptable para otras codificaciones, nombres o formatos de archivos**
- **la separación del frontend y el backend**
- **la claridad del código**
- **el cumplimiento de las reglas de estilo de programación dadas en clase**
- **la eficiencia del procesamiento de los datos y el uso de la memoria.**

## 8. Dudas sobre el TPE

Si bien el enunciado contempla la funcionalidad completa a desarrollar es normal que surjan dudas acerca de cómo interpretar ciertos casos. O que una consigna genere más de una posible solución, por lo que es importante que analicen bien el enunciado, y ante cualquier duda pregunten. Sólo se contestarán dudas sobre el enunciado. Las mismas deben volcarse al **Foro de Discusión “TPE Final” del Campus ITBA**.