

# Trabajo Práctico Especial

## Programación Imperativa

### Segundo Cuatrimestre 2017

#### 1. Objetivo

Diseñar e implementar, en grupos de hasta tres alumnos, un programa para el procesamiento de datos de un censo poblacional. Para ello se deberá realizar tanto el front-end como el back-end, basado en la creación de al menos un TAD.

#### 2. Descripción funcional

El programa deberá leer de entrada estándar líneas de texto hasta leer EOF. En caso de obtener una línea vacía deberá ignorarla. Cada línea tiene un máximo de 80 caracteres.

**Cada una de las líneas representa un habitante registrado en un censo.**

Todas las líneas incluyen datos en formato CSV (datos separados por coma) y tienen el mismo formato. Los datos incluidos, en este orden son:

- **Condición de actividad del habitante**, siendo los valores posibles:
  - 0: Sin Datos
  - 1: Ocupada
  - 2: Desocupada
  - 3: Económicamente inactiva
- **Identificador único del hogar donde habita**. Unsigned Long. Si dos personas censadas tienen el mismo identificador de hogar significa que ambas viven en el mismo hogar.
- **Nombre del departamento** donde habita. String formado por caracteres ASCII estándar (menores a 128)
- **Nombre de la provincia** donde habita. String formado por caracteres ASCII estándar (menores a 128)

Ejemplo de una línea de datos:

3,6546085,Cruz del Eje,Cordoba

donde la persona censada es económicamente inactiva, su hogar tiene el identificador 6546085 y vive en el departamento de Cruz del Eje de la provincia de Cordoba.

Una vez leída toda la información deberá crear los siguientes archivos de texto en la carpeta donde se esté ejecutando el programa, cada uno de ellos en formato CSV, separado por comas:

**1. Pais.csv:** Archivo que contiene una sola línea con los siguientes datos:

- Cantidad de personas que habitan en el país
- Índice de desempleo del país (número real redondeado a dos decimales)

- Cantidad de hogares del país (**Sólo para aquellos que rindan en segunda fecha de final**)

donde el índice de desempleo se calcula a partir de la siguiente fórmula:

$$I_{Desempleo} = \frac{Total\ Desocupados}{(Total\ Ocupados + Total\ Desocupados)}$$

**2. Provincia.csv:** Archivo de texto que contiene una línea por provincia (ordenado en forma alfabética) con los siguientes datos:

- Nombre de la provincia
- Cantidad de personas que habitan en la provincia
- Índice de desempleo de la provincia (número real redondeado a dos decimales)
- Cantidad de hogares de la provincia (**Sólo para aquellos que rindan en segunda fecha de final**)

**3. Departamento.csv:** Archivo de texto que contiene una línea por departamento (donde el orden es primero alfabéticamente por provincia y dentro de cada provincia ordenado alfabéticamente por departamento) con los siguientes datos:

- Nombre de la provincia
- Nombre del departamento
- Cantidad de personas que habitan en el departamento
- Índice de desempleo del departamento (número real redondeado a dos decimales)
- Cantidad de hogares del departamento (**Sólo para aquellos que rindan en segunda fecha de final**)

**4. Provincia.csv (Sólo para aquellos que rindan en segunda fecha de final):** Archivo que contiene una línea por provincia (ordenado en forma alfabética) con los siguientes datos:

- Nombre de la provincia
- Superficie total (en km<sup>2</sup>)
- Densidad de población (en habitantes / km<sup>2</sup>)
- Densidad de hogares (en hogares / km<sup>2</sup>)

donde las densidades mencionadas se calculan a partir de la siguientes fórmulas:

$$D_{Población} = \frac{Total\ Habitantes}{Total\ Superficie}$$

$$D_{Hogares} = \frac{Total\ Hogares}{Total\ Superficie}$$

Para la creación de este archivo necesitarán antes leer de un archivo llamado **superficies.csv** (estará disponible en Campus ITBA) que contiene:

- Nombre de la provincia
- Superficie total en km<sup>2</sup>

## 2.1 Contenidos

Para la realización del presente trabajo se recomienda consultar las siguientes presentaciones en Campus ITBA:

- **Manejo de Archivos:** Para crear los archivos de texto con los resultados del procesamiento de los datos del censo
- **Uso de Valgrind:** Es un conjunto de utilidades que permite encontrar en un programa errores comunes relacionados al manejo de memoria y threads. Para el alcance de este trabajo práctico especial deberán utilizarlo para verificar que el programa a entregar **no presenta leaks de memoria**.

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

### 3. 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.

### 4. Material a entregar.

Cada grupo deberá subir a 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
- makefile (para los que entreguen en segunda fecha)
- Archivos de prueba

### 5. Armado de grupos

Los alumnos deben informar la conformación del grupo y la fecha de final elegida en el Foro de Discusión de Campus ITBA hasta 3 días antes de la fecha de final elegida. También deben inscribirse en la correspondiente fecha de final a través de SGA.

- Para los que rindan en primer fecha: el trabajo debe subirse a Campus antes de las 12:00 hs (mediodía) del lunes 11/12.
- Para los que rindan en segunda fecha: el trabajo debe subirse a Campus antes de las 12:00 hs (mediodía) del lunes 18/12.

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.

### 6. 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, 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 correcto funcionamiento del programa (recordar el uso de métodos de prueba de software)
- la modularización realizada
- el correcto diseño del TAD, que debería poder ser utilizado para censos de otros países, y 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.

## **7. 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. Las dudas sobre enunciado como así también sobre los temas a estudiar (archivos, makefile, etc.) DEBEN ser hechas exclusivamente a través del foro “Dudas TPE” del Campus ITBA. Y se asume que toda respuesta y aclaración hecha por la Cátedra como respuesta a las mismas es leída y tomada en cuenta por TODOS los grupos.