

## Programación 2

### Práctico 0 - Introducción a C\*

#### Objetivos

- Repasar conceptos de programación adquiridos en Programación 1, aplicando éstos en el lenguaje de programación C\* que emplea funcionalidades de C y C++.
- Familiarizarse con el compilador y con el entorno de programación.

#### Comentarios

Los ejercicios deben ser resueltos en C\* que tiene funcionalidades básicas provenientes de C y C++. Se sugiere que previo a la resolución de los ejercicios se consulte la bibliografía referente al lenguaje que se menciona en el sitio del curso.

#### Ejercicio 1

Suponga que a los estudiantes de programación 2 se les dice que su calificación final será el promedio de las cuatro calificaciones más altas de entre las cinco que hayan obtenido en el curso.

(a) Escribir una función llamada PromClase con cinco parámetros de entrada (las calificaciones obtenidas) y un parámetro de salida (la calificación promedio), que realice dicho cálculo

(b) Escribir un programa principal main() que permita ejecutar la función PromClase. Dicho programa deberá leer de la entrada estándar (teclado) 5 calificaciones, invocar al procedimiento PromClase con dichos parámetros, finalmente mostrar en la salida estándar (pantalla) el resultado.

#### Ejercicio 2

Escriba un procedimiento que calcule e imprima en pantalla todos los números primos entre dos enteros positivos A y B cualesquiera.

#### Ejercicio 3

Se quiere implementar una función que cuente la cantidad de Veces que una letra aparece en una frase. La frase se representa como un arreglo de caracteres, y dado que se conoce que el largo máximo de una frase es de 100 caracteres, la frase se implementa como `char frase[100]`.

Usando esta representación escriba una función Ocurrencias que recibe una frase, un natural llamado largo que representa el número de caracteres en la frase, y el carácter a buscar (almacenado en la variable letra), y devuelve el número de ocurrencias del carácter letra en el arreglo frase.

#### Ejercicio 4

Considere ahora que la frase se representa como un arreglo de caracteres implementado como `char *frase`.

- (a) ¿que diferencias hay entre esta representación y la que utilizó en el Ejercicio 3?
- (b) Escriba una función `EsPalindrome` que recibe una frase representada como un puntero a `char` y devuelve `TRUE` si la misma es un palíndromo, `FALSE` en otro caso.

#### Ejercicio 5

Escriba un procedimiento que recibe un arreglo de enteros y devuelve un nuevo arreglo que contiene a los elementos del primero en orden ascendente. Indique que algoritmos de ordenación utiliza.

#### Ejercicio 6

Considere una matriz de enteros representada como un arreglo bidimensional de diez filas y diez columnas.

- (a) Escriba un procedimiento llamado `ImprimirMatriz`, que recibe como parámetro una matriz e imprime su contenido en pantalla, imprimiendo sus elementos separados por un espacio y cada fila en una línea distinta.
- (b) Escriba un procedimiento llamado `Cambio`, que recibe como parámetros una matriz y dos variables naturales `m` y `n` y modifica la matriz parámetro, intercambiando sus filas `m` y `n`.

#### Ejercicio 7

Considere ahora que se desea representar una matriz de enteros de `M` filas y `N` columnas mediante un arreglo de enteros de tamaño `M*N` (Ver la Figura 1 a modo de ejemplo). Implemente las siguientes operaciones utilizando la representación mencionada:

- (a) `Crear`: recibe dos enteros `M` y `N` y devuelve un arreglo de tamaño `N*M`
- (b) `Asignar`: recibe un arreglo `m`, un Valor `X` y dos índices `i` y `j`, y asigna el Valor `X` en el lugar `(i,j)` considerando que el arreglo `m` representa una matriz
- (c) `Valor`: recibe un arreglo `m` y dos índices `i` y `j`, devuelve el Valor en el lugar `(i,j)` considerando que el arreglo `m` representa una matriz
- (d) `Traspuesta` : recibe un arreglo `m` que representa una matriz y devuelve un nuevo arreglo `n` que contiene a la matriz traspuesta de la representada en `m`

$$A_{2,3} = \begin{bmatrix} 3 & 2 & 5 \\ 1 & 0 & 10 \end{bmatrix} \quad \boxed{3 \mid 2 \mid 5 \mid 1 \mid 0 \mid 10}$$

**Figura 1:** Ejemplo de representación de una matriz de 2 filas y 3 columnas utilizando un arreglo unidimensional de largo 6