

**TÉCNICAS DE PROGRAMACIÓN**  
**CUARTO LABORATORIO CALIFICADO**  
**SEMESTRE ACADÉMICO 2023-2**

Horarios: Todos

Duración: 110 minutos  
Elaborado por los profesores del curso.

**ADVERTENCIAS:**

- Todo dispositivo electrónico (teléfono, tableta, computadora u otro) deberá permanecer apagado durante la evaluación **en su mochila**.
- Coloque todo aquello que no sean útiles de uso autorizado durante la evaluación en la parte delantera del aula, por ejemplo, mochila, maletín, cartera o similar, y procure que contenga todas sus propiedades. La apropiada identificación de las pertenencias es su responsabilidad.
- Si se detecta omisión a los dos puntos anteriores, la evaluación será considerada nula y podrá conllevar el inicio de un procedimiento disciplinario en determinados casos.
- Es su responsabilidad tomar las precauciones necesarias para no requerir la utilización de servicios higiénicos: durante la evaluación, no podrá acceder a ellos, de tener alguna emergencia comunicárselo a su jefe de práctica.
- Quienes deseen retirarse del aula y dar por concluida su evaluación no lo podrán hacer dentro de la primera mitad del tiempo de duración destinado a ella.

**INDICACIONES:**

- No se pueden usar apuntes de clase ni calculadoras.
- Está prohibido el acceso a Internet y a correo electrónico hasta que lo indiquen los jefes de práctica, tampoco podrá emplear dispositivos USB.
- Si se detecta omisión al punto anterior, la evaluación será considerada nula y podrá conllevar el inicio de un procedimiento disciplinario en determinados casos.
- LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE, por lo que NO SE CALIFICARÁN aquellos módulos que son llamados por otros que estén incompletos. Cada módulo no debe sobrepasar las 30 líneas de código aproximadamente.
- NO SE PUEDEN EMPLEAR ARCHIVOS DE DATOS AUXILIARES NI VARIABLES GLOBALES. NO podrá implementar funciones en el archivo main.cpp, las funciones se deberán implementar en archivos independientes (.h y .cpp).
- En la calificación se tomará en cuenta el buen uso de los nombres de los identificadores, y el eficaz uso de comentarios.
- **DEBE COLOCAR SU NOMBRE Y CÓDIGO EN EL ARCHIVO main.cpp DE SU PROYECTO, DE LO CONTRARIO SE LE DESCONTARÁ 0.5 PUNTOS EN SU NOTA FINAL. NO SE HARÁN EXCEPCIONES.**
- **NO PUEDE TENER EN SU NetBeans OTROS PROYECTOS ABIERTOS AL INDICADO EN ESTE LABORATORIO**
- **NO PODRÁ EMPLEAR ARREGLOS DE CADENAS DE CARACTERES EN ESTE LABORATORIO. TAMPOCO PODRÁ EMPLEAR LAS BIBLIOTECAS <stdio.h>, <cstdlib>, <string.h>, <cstring> ni <string>.**

**INDICACIONES INICIALES**

Siga estrictamente las indicaciones que a continuación se detallan:

- En la unidad de trabajo indicada por los Jefes de práctica (**si trabaja en otra unidad, no se calificará su laboratorio y se le asignará como nota cero**), cree allí una carpeta con el nombre **"CO\_PA\_PN\_Lab04\_2023\_2"** donde **CO** indica: Código del alumno, **PA** indica: Primer Apellido del alumno y **PN** primer nombre (de no colocar este requerimiento se le descontará 3 puntos de la nota final). **Allí colocará los dos proyectos solicitados en la prueba. NO SE HARÁN EXCEPCIONES SI NO ACATA ESTAS INDICACIONES.**

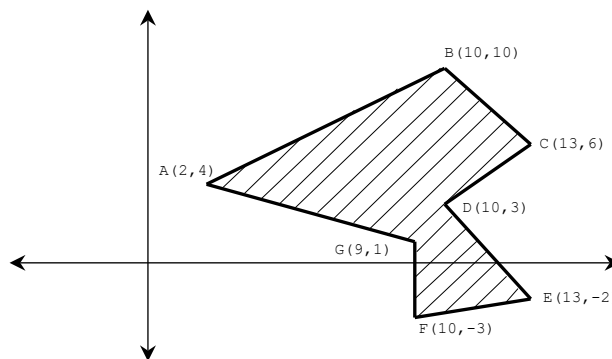
La finalidad de este laboratorio es la de reforzar los conceptos y el uso de arreglos.

**DEBE LEER TODA LA PRUEBA ANTES DE EMPEZAR A DESARROLLAR EL PROGRAMA**

**PREGUNTA 1 (10 puntos)**

Cree un proyecto en NetBeans con el nombre: **"Arreglos\_Area\_Y\_Perimetro\_Lab\_04\_2023-2"** (de no respetar este nombre se le descontarán 1 punto de su nota final – **NO SE HARÁN EXCEPCIONES**) y en él desarrolle el programa que resuelva el problema que se describe a continuación.

Dadas las coordenadas de un polígono cualquiera, un algoritmo que permite determinar el área de un polígono se describe a continuación:



Se toman las coordenadas en sentido horario y se operan de la siguiente manera:

La primera coordenada se vuelve a usar al final

	X	Y
A	2	4
B	10	10
C	13	6
D	10	3
E	13	-2
F	10	-3
G	9	1
A	2	4

→  $10 \times 4 + 13 \times 10 + 10 \times 6 + 13 \times 3 + 10 \times (-2) + 9 \times (-3) + 2 \times 1 \rightarrow S1=224$

Es decir:  $S1 = \sum_{i=1}^n X_i \cdot Y_{i-1}$

	X	Y
A	2	4
B	10	10
C	13	6
D	10	3
E	13	-2
F	10	-3
G	9	1
A	2	4

→  $10 \times 2 + 6 \times 10 + 3 \times 13 + (-2) \times 10 + (-3) \times 13 + 1 \times 10 + 4 \times 9 \rightarrow S2=106$

Es decir:  $S2 = \sum_{i=1}^n Y_i \cdot X_{i-1}$

El área se determinará con la siguiente fórmula:  $\frac{S1-S2}{2}$

Esto es:  $\text{Área} = \frac{(224-106)}{2} = 59.0$

Para el cálculo del perímetro, considerar que la longitud de un segmento de recta entre dos puntos se calcula con la fórmula siguiente:  $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Se pide que elabore un programa en lenguaje C++ que lea de un archivo de textos, las coordenadas (x, y) de un polígono y a partir de allí determine el número de lados, el área y el perímetro del polígono. Los valores para las coordenadas serán de punto flotante. Las coordenadas en el archivo estarán dadas en el archivo en sentido horario. Considerar un máximo de 20 vértices.

Las tareas que realizará este proyecto son:

- Leer las coordenadas del polígono y colocarlas en arreglos.
- Probar que los arreglos tengan los datos correctos, imprimiendo un reporte sencillo (cada columna debe tener un título apropiado) y los datos deben estar correctamente alineados.
- Calcular el área, perímetro y número de lados del polígono.
- Mostrar las coordenadas y resultados en un reporte bien alineado, con títulos y etiquetas adecuados.

## PREGUNTA 2 (10 puntos)

Cree un proyecto en NetBeans con el nombre: “Arreglos\_Picos\_Y\_Valles\_Lab\_04\_2023-2” (de no respetar este nombre se le descontarán 1 punto de su nota final – NO SE HARÁN EXCEPCIONES) y en él desarrolle el programa que resuelva el problema que se describe a continuación.

Se tiene un archivo de textos con las ganancias de una compañía a través de los años, el archivo es similar al que se muestra en la figura No. 1.

1887	91	1888	88	1889	64	1890	11
1891	2	1892	92				
1893	67	1894	50	1895	48		
1896	14						
1897	3	1898	10	1899	17	1900	52
1901	69	...					
...							

Cada par de datos indica el año del registro y la ganancia que se obtuvo ese año en millones de dólares.

Figura No. 1

Si graficáramos esos datos, obtendríamos una figura como la figura 2

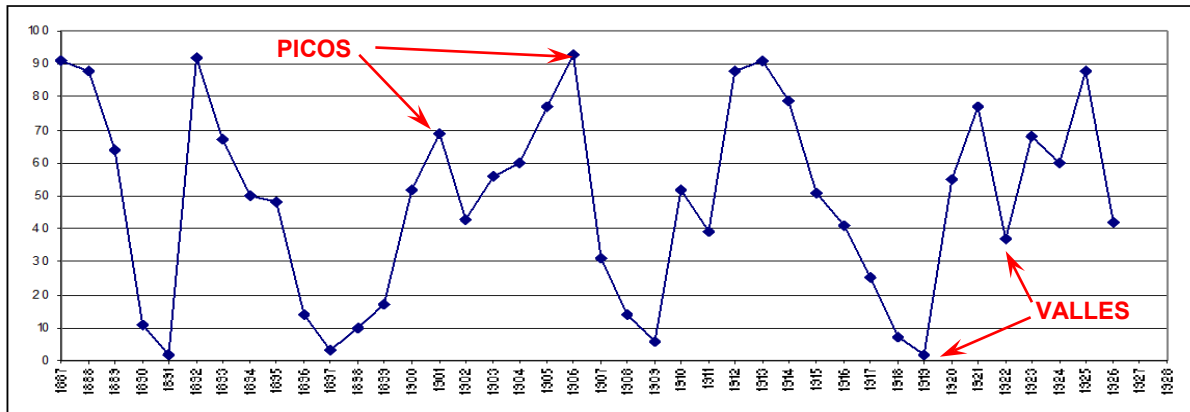


Figura No. 2

Se desea que usted elabore un programa en C++ que permita leer los datos antes mencionados luego imprima los años en que se produjeron picos y los años en que se produjeron valles, de la siguiente manera:

EMPRESA MULTINACIONAL TP GANANCIAS A TRAVÉS DEL TIEMPO	
=====	
RELACION DE PICOS:	
Periodo	Ganacia
-----	
1887	91
1892	93
...	...
=====	
PICO MAYOR: 1906 con 95 millones	
=====	
RELACION DE VALLES:	
Periodo	Ganacia
-----	
1891	2
1897	4
...	...
=====	
VALLE MENOR: 1919 con 1 millon	

Las tareas que realizará este proyecto son:

- Leer los datos de años y ganancias del archivo y cargarlos en arreglos.
- Probar que los arreglos tengan los datos correctos, imprimiendo un reporte sencillo (cada columna debe tener un título apropiado) y los datos deben estar correctamente alineados.
- Emitir un reporte, como se muestra en el ejemplo, indicando todos los picos y valles e identificando al pico mayor y al valle menor.

Considerar un máximo de 100 juegos de datos.

### **CONSIDERACIONES IMPORTANTES A TENER EN CUENTA EN TODO EL LABORATORIO:**

- Los archivos solo pueden leerse una vez.
- Cada tarea debe ser desarrollada en una función independiente. Puede definir sub tareas (funciones) para resolver alguna tarea indicada en la pregunta. Si emplea una función para desarrollar dos o más tareas indicadas, no se calificará el proceso.
- Debe usar obligatoriamente la biblioteca fstream para la entrada y salida de datos.

**Al finalizar el laboratorio, comprima la carpeta de su proyecto empleando el programa Zip que viene por defecto en el Windows, no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares y súbalo a la tarea programa en Paideia para este laboratorio.**

#### **ADVERTENCIAS:**

Obligatoriamente debe desarrollar su proyecto bajo NetBeans en Windows, no podrá desarrollarlo empleando otro IDE ni otro sistema operativo.

**CRITERIOS DE CALIFICACIÓN:**

1. Si el programa entregado presenta más de tres errores de sintaxis serán calificados sobre la mitad del puntaje.
2. Si el programa no muestra los resultados o los muestren y no sean correctos, no podrán tener más del 75% de la nota.
3. Se descontará 15% de la nota si el programa define variables con nombres que no tengan sentido. Las variables deben empezar con una minúscula, se emplearán mayúsculas para separar las palabras compuestas (p. e.: baseInferior).
4. Se descontará 15% de la nota si no se colocan comentarios relevantes, incluyendo un encabezado al inicio del programa en el que se indique el nombre del autor, la fecha, y una descripción de lo que hace programa).
5. No se calificará el código puesto como comentario.
6. No se calificarán aquellas funciones implementadas en el archivo main.cpp

San Miguel, 25 de septiembre del 2023