

ALUMNO:

EPS -Ingeniería Informática

Asignatura: G0460006 Programación I – TURNO 3

Curso: 2022/2023
Semestre: 1º

Examen: Final
Convocatoria: Extraordinaria

Fecha: 26-6-2023 (15:00-18:00)

Parte Práctica (10 puntos; 70% nota final) - Tiempo: 2h40m

Los ficheros correspondientes a los ejercicios se deben entregar en la actividad correspondiente a través del campus antes de la finalización del tiempo establecido. Cada fichero se llamará EjercicioX, donde la X será el número de dicho ejercicio. Los únicos formatos válidos serán **txt** o **cpp**, siempre **sin comprimir**.

Criterios generales de evaluación

Funciones/Métodos: Si no se usa el paso por referencia constante cuando las variables de los parámetros de entrada no son de tipo simple.	40%
Tipos de datos y variables: <ul style="list-style-type: none">• Uso de variables globales (fuera del ámbito de una función).• Si no se usan los tipos contenedor vistos en clase (std::array; std::vector; std::set; std::string, etc.) para las variables que lo necesiten.• Si no se usan punteros inteligentes (std::unique_ptr; std::shared_ptr) cuando sea necesario	0% 0% 0%
El programa no compila o no se asemeja a lo pedido.	0%
Si no se cumplen los criterios de entrega indicados en la actividad/examen .	0%

Criterios particulares de evaluación

El elemento evaluable no compila o no se asemeja a lo que se pide	0%
El elemento evaluable no se aproxima suficientemente a lo pedido	40%
El elemento evaluable se aproxima suficientemente a lo pedido	60%
El elemento evaluable funciona correctamente y las estrategias y elementos de código elegidos son adecuados.	100%

IMPORTANTE:

- Todos los ejercicios del examen deberán ser resueltos de forma **algorítmica**, es decir, la **solución** propuesta tendrá que ser **general** y **no particular** para unos determinados datos/valores.
- Todos los ejercicios resueltos sin utilizar funciones cuando sea apropiado se valorarán con una nota máxima del 60% sobre la calificación prevista.
- Se recomienda una primera lectura del examen completo para planificar la realización del examen. Y una segunda lectura detallada antes de la realización de cada uno de los ejercicios propuestos.

Ejercicio 1 (3 puntos)

Escriba un programa en C++11, *Ejercicio1*, utilizando los tipos de datos, las **estructuras de control** y las **funciones** necesarias para determinar la **suma de polinomios de una variable de grado máximo 10**.

Definimos **polinomio de una variable**, $P(x)$, como una expresión formada por la suma de varios términos (monomios) donde cada uno es el producto de un **coeficiente de tipo real (float)** y una variable x elevada a un **exponente de tipo entero (int)**. Así, el **grado del polinomio** $P(x)$ es el mayor exponente al que se encuentra elevada la variable x .

Ejemplos: $P1(x) = 2.6x^3 - 5.8x + 7.2$, polinomio de **grado tres**.

$P2(x) = 4.5x^4 + 3.2x^3 - 7.1x^2 - 1.6$, polinomio de **grado cuatro**.

$P3(x) = -8.2x^5 + 4.5x^4 + 2.4x^2 - 3.0x + 1.9$, polinomio de **grado cinco**.

El proceso deberá realizar las siguientes tareas:

- Almacenar dos polinomios utilizando **std::array** [**0,75 puntos**]. Por ejemplo, el $P3$ quedaría:

1.9	-3.0	2.4	0	4.5	-8.2	0	0	0	0	0
-----	------	-----	---	-----	------	---	---	---	---	---

 $P3(x)$

- Determinar su suma de dos polinomios cualesquiera $P_s(x) = P1(x) + P2(x)$ [**1 punto**].

7.2	-5.8	0	2.6	0	0	0	0	0	0	0
-----	------	---	-----	---	---	---	---	---	---	---

 $P1(x)$

-1.6	0	-7.1	3.2	0	4.5	0	0	0	0	0
------	---	------	-----	---	-----	---	---	---	---	---

 $P2(x)$

5.6	-5.8	-7.1	5.8	0	4.5	0	0	0	0	0
-----	------	------	-----	---	-----	---	---	---	---	---

 $P_s(x)$

- Mostrar por pantalla, tanto los polinomios sumandos como el polinomio suma. Los polinomios se deben mostrar de forma ordenada (creciente o decreciente) según los exponentes de la variable x , **mostrando únicamente los términos no nulos** (coeficientes distintos de cero) [**1,25 puntos**].

Por ejemplo, en pantalla debe aparecer: $P1(x) = + 7.2 - 5.8x^1 + 2.6x^3$

$P2(x) = - 1.6 - 7.1x^2 + 3.2x^3 + 4.5x^5$

$PS(x) = + 5.6 - 5.8x^1 - 7.1x^2 + 5.8x^3 + 4.5x^5$

Ejercicio 2 (3 puntos)

Escriba un programa en C++11, *Ejercicio2*, utilizando los tipos de datos, las **estructuras de control** y las **funciones** necesarias para calcular y almacenar en **una sola variable**, del tipo más adecuado, los **20 primeros términos** de la siguiente serie: **[1 punto]**

$$A_n = -3 A_{n-2} + 2 A_{n-1} \text{ siendo } A_0 = 1 \text{ y } A_1 = -1 \quad \forall n \geq 2$$

A continuación, partiendo de la variable que contiene los valores de la serie:

- Crear **dos variables** que permitan almacenar, por un lado, el **menor valor positivo**, el **mayor valor positivo** y todos los **valores positivos** de los términos de la serie (incluidos los valores nulos o ceros, 0) y, por otro lado, el **menor valor negativo**, el **mayor valor negativo** y todos los **valores negativos** de los términos de la serie. **[1,5 puntos]**
- Mostrar por pantalla **todas las variables** creadas en el programa. **[0,5 puntos]**

Ejercicio 3 (4 puntos)

Escriba un programa en C++11, *Ejercicio3*, utilizando los tipos de datos, las **estructuras de control** y las **funciones** necesarias para solicitar al usuario una frase (tipo `std::string`) que debe ser leída por teclado y almacenar en **una sola variable**, del tipo que consideres más adecuado, la información que se obtiene después de realizar sobre la frase las siguientes tareas **a través de una función que devuelva un puntero inteligente a la variable definida:**

- Determinar **cuántas veces aparece cada una de las vocales**, debiendo contabilizar tanto si aparecen en **minúscula** como en **mayúscula** dentro de la frase **(1,5 puntos)**.
- Determinar en **qué posiciones dentro de la frase están cada una de las vocales diferenciando de cuál se trata**, es decir, de la vocal ‘a’, vocal ‘e’, etc. **(1,5 puntos)**.

Si la función implementada en lugar de devolver un puntero inteligente devuelve únicamente la variable con la información calculada, este apartado puntuará como máximo [1,5 puntos] en lugar de [3 puntos]

Finalmente, realizar una función que **reciba como argumentos la frase y el puntero inteligente a la variable donde se ha guardado todo lo calculado en el proceso** para mostrar por pantalla todo su contenido **[1 punto]**.

En caso de no haber implementado la versión con punteros este apartado puntuará como máximo [0,5 puntos] en lugar de [1 punto]

Por ejemplo: para la frase “**Si no puedo dibujarlo, es que no lo entiendo**”, deberá indicarse por pantalla lo siguiente:

Las vocales de la frase - **Si no puedo dibujarlo, es que no lo entiendo** son:

Letra a/A: 1 en la/s posicion/es: 17

Letra e/E: 5 en la/s posicion/es: 8, 23, 28, 36, 40

Letra i/I: 3 en la/s posicion/es: 1, 13, 39

Letra o/O: 6 en la/s posicion/es: 4, 10, 20, 31, 34, 43

Letra u/U: 3 en la/s posicion/es: 7, 15, 27