

Indicaciones específicas:

- Esta evaluación contiene 5 páginas (incluyendo esta página) con 2 preguntas. El total de puntos son 20.
- El tiempo límite para la evaluación es 100 minutos.
- Cada pregunta deberá ser respondida en un solo archivo con el número de la pregunta.
 - p1.cpp
 - p2.cpp
- Deberás subir estos archivos directamente a www.gradescope.com, uno en cada ejercicio. También puedes crear un .zip

Competencias:

- Para los alumnos de la carrera de Ciencia de la Computación
 - Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (Evaluar)
 - Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución.(Usar)
 - Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (Usar)
- Para los alumnos de las carreras de Ingeniería
 - Capacidad de aplicar conocimientos de matemáticas (nivel 3)
 - Capacidad de aplicar conocimientos de ingeniería(nivel 2)
 - Capacidad para diseñar un sistema, un componente o un proceso para satisfacer las necesidades deseadas dentro de restricciones realistas (nivel 2)

Calificación:

Tabla de puntos (sólo para uso del professor)

Question	Points	Score
1	13	
2	7	
Total:	20	

1. (13 points) **Clases y Objetos**

Implementar la clase **Polinomio** con miembros privados a_0, a_1, \dots, a_n . Estos miembros representaran los coeficientes de un polinomio de grado n de la forma:

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0.$$

Para ello, el programa debe cumplir con los siguientes requisitos:

- En la ejecución, el programa debe solicitar el grado de dos polinomios (pueden ser de distinto grado). Luego crear los coeficientes con números enteros aleatorios (positivos, negativos o cero).
- Realizar la sobrecarga del operador '<<' para visualizar objetos **Polinomio**. Por ejemplo, si fuera un polinomio cuadrático, se debería visualizar en el siguiente formato:

$$ax^2 + bx + c$$

Donde **a**, **b** y **c** son los coeficientes del objeto **Polinomio**. Además, no mostrar los términos a_n, a_{n-1}, \dots, a_1 si son cero. En ejemplo anterior, los términos **a** o **b** no deberían visualizarse si son cero. Además, si un coeficiente tiene signo negativo debería de suprimirse el símbolo '+'.

- Realizar la sobrecarga del operador '+' para retornar la suma de dos objetos **Polinomio**.
- En la ejecución, el programa debe mostrar algún ejemplo de que efectivamente se cumplen los requisitos anteriores.

Algunos ejemplos de diálogo de este programa serían:

Listing 1: Ejemplo 1

```
// p1.cpp

Polinomio sum = poli1 + poli2;
cout << poli1 << ":poli1" << endl;
cout << poli2 << ":poli2" << endl;
cout << sum << ":poli1+poli2" << endl;;
```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Clases y Objetos	Desarrolla las clases con todos los atributos y métodos requeridos, se realiza un adecuado encapsulamiento, se utiliza adecuadamente los constructores que permiten copiar y asignar objetos, se realiza las sobrecargas adecuadas, se realiza un uso adecuado de la memoria dinámica (13pts)	Desarrolla las clases con todos los atributos y métodos requeridos, se realiza un adecuado encapsulamiento, se realiza las sobrecargas adecuadas, se realiza un uso adecuado de la memoria (9pts)	Desarrolla las clases con todos los atributos y métodos requeridos, se realiza las sobrecargas adecuadas, errores en el funcionamiento pasa algunas pruebas (5pts).	Contiene errores que no hace que funcione el programa (1pts)

2. (7 points) *Variadic Templates y Fold Expressions*

Implemente la función:

$$\text{foo}(x_1, x_2, x_3, \dots, x_n)$$

que retorne el valor de:

$$x_1 - x_2 - x_3 - \dots - x_n$$

Algunos ejemplos de diálogo de este programa serían:

Listing 2: Ejemplo 1

```
// p2.cpp

cout << foo(2) << endl; // resultado = 2
cout << foo(1,12) << endl; // resultado = 1-12 = -11
cout << foo(7,3,0.5,1,2) << endl; // resultado = 0.5
```

La rúbrica para esta pregunta es:

Criterio	Excelente	Adecuado	Mínimo	Insuficiente
Template de Funciones	Buen nivel de abstracción, uso adecuado de los diferentes parámetros de templates, selección correcta del tipo de template, selección correcta de la especialización o recursividad de templates. (7pts)	Buen nivel de abstracción, uso adecuado de los diferentes parámetros de templates. Selección correcta del tipo de template, uso de especializaciones o recursividad. (5pts)	Buen nivel de abstracción, uso adecuado de los diferentes parámetros de templates. Selección correcta del tipo de template, errores en el funcionamiento pasa algunas pruebas. (3pts).	No se hizo un uso adecuado de los templates. (1pts)