

Facultad de Matemática y Computación (UH)
Ciencia de la Computación
Matemática Numérica
Curso 2024

Clase Práctica # 1.2246467991473532e-16:
Tutorial/Introducción a las clases prácticas

Esta clase práctica es un tutorial a las clases prácticas de la asignatura. Por ese motivo aparecerán algunos textos como este a lo largo del documento. En el resto de las CP no habrá tanto texto como en esta.

Lo primero es que las CP están divididas en secciones con nombre como la siguiente:

Ejercicios muuuy fáciles.

Usualmente los ejercicios en la primera sección de la CP son tan fáciles que se pueden hacer sin mucho problema. De hecho, están pensados para que todo el mundo los pueda hacer, o que pase mucho trabajo para inventarse una excusa de por qué no los hizo.

Ejercicio 1: Notación científica en la computadora. (10000¹ créditos)

Escribe los siguientes números utilizando **exactamente** la cantidad de caracteres indicados, de forma que se puedan reconocer como números en el lenguaje de programación de su preferencia².

- a) 4500000, con 4 caracteres: _ _ _ _
- b) -1230000, con 6 caracteres: _ _ _ _ _ _
- c) 0.000000123, con 6 caracteres: _ _ _ _ _ _
- d) 1, con 4 caracteres: _ _ _ _

¹En el nombre de cada ejercicio está indicada la cantidad de créditos que se reciben por entregar el ejercicio resuelto. Si un ejercicio no se hace completo, la cantidad de créditos que se recibe es proporcional al fragmento del ejercicio que fue respondido.

²Se supone que la mayoría de las actividades del curso las vamos a hacer con Python, pero en algunos casos (como este), se puede usar cualquier otro lenguaje, y en otros, se debe usar un lenguaje específico, pero eso se indicará en las orientaciones.

Ejercicio 2: ¡No confíes en los cálculos de las computadoras! (40000 créditos)

En el lenguaje de programación de su preferencia:

- a) Diga cuáles de estas igualdades (o desigualdades) son verdaderas (*3000 créditos cada una*).

a) $0.4 * 6 > 2.4$

b) $0.8 * 3 == 0.3 * 8$

c) $0.3 * 3 == 0.9$

d) $3.1 * 2 < 6.2$

e) $1e100 + 1e50 == 1e100$

- b) (*10000 créditos*) Determine cuál es el menor valor natural N para el cual se cumple que:

$$1e100 + 10^N > 1e100.$$

- c) (*10000 créditos*) En algunos lenguajes de programación, como **C#** o **C**, es muy fácil trabajar con distintos tipos de números, por ejemplo, **float** y **double**. Resuelva el siguiente ejercicio para el tipo de datos **float** y para el tipo de datos **double**:

$$1e24 + 10^N > 1e24.$$

- d) (*5000 créditos*) ¿Qué ocurre si se trata de resolver el inciso b) con el tipo de dato **float**?

Ejercicio 3: Python! (50000 créditos)

El objetivo de este ejercicio es tener una primera familiarización con el lenguaje de programación Python, y una buena forma de hacerlo es implementado algo que sea fácil, por ejemplo, el factorial de un número :-D. Por eso, lo que hay que hacer es una función en Python que reciba un número entero y devuelva su factorial. Para ello:

- a) (*10000 créditos*) Haga una implementación recursiva.
- b) (*10000 créditos*) Haga una implementación iterativa.
- c) (*10000 créditos*) Haga una implementación que use la función Gamma que está implementada en el módulo **scipy**.
- d) (*10000 créditos*) Escriba una variante de la versión iterativa en la que, aunque el usuario pase el argumento como entero, los cálculos se realicen con números flotantes³
- e) (*10000 créditos*) ¿A partir de qué valor de N los resultados de la variante con enteros es distinta de la variante con flotantes?⁴

³Probablemente sea suficiente con poner que el valor inicial del resultado sea 1.0 en vez de 1.

⁴¿Qué cosa más loca! Pero sí :-), no es lo mismo el factorial con flotantes que con enteros :-o.

Ejercicio 4: Análisis Matemático en Python. (70000 créditos)

La definición de derivada de una función $f : \mathbb{R} \rightarrow \mathbb{R}$ en un punto \bar{x} es

$$f'(\bar{x}) = \lim_{h \rightarrow 0} \frac{f(\bar{x} + h) - f(\bar{x})}{h}.$$

Una manera de aproximar la derivada de una función $f(x)$ en un punto \bar{x} es:

$$f'_h(\bar{x}) \approx \frac{f(\bar{x} + h) - f(\bar{x})}{h}$$

donde h es un valor pequeño.

- Implemente en Python una función que devuelva una aproximación de la derivada de una función dada, en un punto \bar{x} usando un parámetro h . La función implementada debe recibir tres argumentos `f`, `x`, y `h`, donde `f` es una función $f : \mathbb{R} \rightarrow \mathbb{R}$ implementada en Python, `x` es un número real donde se quiere aproximar la derivada de f , y `h` es el valor del parámetro presente en la aproximación propuesta.
- Verifique el buen funcionamiento de su implementación aproximando la derivada de la función $f(x) = x^2$, en el punto $\bar{x} = 1$, con varios valores de h .
- ¿Para qué valor de h se obtienen los mejores resultados en el inciso anterior?⁵

Ejercicio 5: Primera introducción a la Serie de Taylor. (50000 créditos)

Dados una función $f \in C^\infty$, un punto $x \in \text{Dom} f$, y $h \in \mathbb{R}$ se puede aproximar el valor $f(x_0 + h)$ mediante la expresión:

$$f(x_0 + h) = f(x) + \frac{f'(x_0)h}{1!} + \frac{f''(x_0)h^2}{2!} + \frac{f'''(x_0)h^3}{3!} + \dots$$

- ¿Cómo se llama esta forma de aproximar una función?⁶
- ¿Cuál es la expresión general para este tipo de aproximación?
- Calcule el desarrollo en serie de _____⁷ de las siguientes funciones, si se desarrollan alrededor del punto $x_0 = 0$:
 - $f(x) = e^x$
 - $f(x) = \text{sen}(x)$
 - $f(x) = x^5 + 6x^3 - 4x^2 + 5$

⁵¡Taratatáaan! Acabas de descubrir una pregunta secreta por un valor de 30000 créditos. *Voz del tutorial: En algunas clases prácticas hay ejercicios secretos como este que acabas de encontrar. Siempre puedes distinguir una pregunta secreta de una nota al pie tradicional sonido que hacen al ser leídas, que es ¡Taratatáaan! En el caso de las preguntas secretas, para ganar los créditos solo tienes que responderlas normalmente (si puedes :-P). En el caso de esta ¿primera? pregunta secreta de esta clase práctica lo que debes responder es: ¿Cómo se puede saber si la aproximación con un valor de h es mejor que con otro?*

⁶*Voz de tutorial: En muchos ejercicios de la asignatura aparecen pistas que se pueden usar, por ejemplo en el nombre del ejercicio, o en otros lugares del mismo.*

⁷Este espacio en blanco debe ser rellenado con la respuesta al inciso a)

Ejercicio 6: ¿Has visto las series de Taylor? (50000 créditos)

Para cada una de las funciones del ejercicio anterior, escriba una función en Python que reciba un número real x , un entero n menor o igual que 3, los extremos de un intervalo $[a, b]$ y usando `matplotlib` construya un gráfico en el intervalo $[a, b]$ en el que se muestre:

- la función original y
- el polinomio de Taylor de grado n alrededor del punto x .

Ejercicios más interesantes

Por lo general los ejercicios en esta segunda sección son más interesantes que los de la primera. Por ejemplo, rara vez son solo mecánicos, y te permitirán combinar conocimientos y habilidades de una manera creativa y elegante, y son el tipo de ejercicios que si durante el curso los hiciste todos, podrás terminar la asignatura cómodamente con 4 puntos.

Ejercicio 7: En esta asignatura el tamaño⁸ sí importa. (30000 créditos)

La derivada de una función $f : \mathbb{R} \rightarrow \mathbb{R}$ se puede aproximar de varias formas. Algunas de ellas son:

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} \quad (1)$$

y

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \quad (2)$$

Tomando como ejemplo las funciones $f_1(x) = x^2$ y $f_2(x) = x^3$, responda las siguientes preguntas:

- a) Si se usa un valor de $h = 0,1$ en las aproximaciones (1) y (2) para calcular las derivadas de f_1 y f_2 en el punto $\bar{x} = 1$, ¿**exactamente** qué error se comete en cada caso?

Ejercicio 8: Lo que tus profesores de Álgebra no querían que supieras (50000 créditos)

Uno de los temas del curso del álgebra son las operaciones con matrices. En este ejercicio es una introducción a `numpy`, una biblioteca de Python que permite realizar, con mucha facilidad, operaciones con matrices, de esas que en primer año eran muy tediosas. Para ello lo que se debe hacer es:

- a) Escriba una función en Python que reciba tres matrices y, usando `numpy`, determine si el producto de las dos primeras “es igual” a la tercera.

⁸(Del error)

- b) Escriba una función en Python, que usando la función implementada en el inciso *a*), reciba una matriz A y dos vectores \mathbf{x}_1 y \mathbf{b} (con las dimensiones correspondientes) y determine si el vector \mathbf{x}_1 es solución del sistema $Ax = b$.

Ejercicio 9: Cuando a la computadora no le gusta tu álgebra (50000 créditos)

Dados la matriz $A_n = \alpha_{i,j}$, donde $\alpha_{i,i} = 0,5$, para todo $i = \overline{1, n}$, $\alpha_{i,i+1} = 1$, para todo $i = \overline{1, n-1}$, y $\alpha_{i,j} = 0$ en otro caso; y el vector columna $\mathbf{1}_n$, que tiene dimensión n y un 1 en cada componente:

- a) (20000 créditos) Implemente una función en Python que reciba un parámetro n , y devuelva un vector x_e que sea la solución **exacta**⁹ del sistema:

$$A_n x_e = \mathbf{1}_n.$$

- b) (20000 créditos) Usando alguna de las funciones implementadas en el ejercicio 8, reciba un número n y verifique si el vector x que devuelve la función implementada en el inciso *a*) es solución del sistema $A_n x = \mathbf{1}_n$.
- c) (10000 créditos) Use la función implementada en el inciso *b*) y compruebe que la propuesta implementada en el inciso *a*) funciona para los siguientes valores de n : 20, 40, 60, 80 y 100.

Ejercicio 10: No hagas caso a los rumores (a no ser que sean ciertos :-/) (75000 créditos)

A cada rato hay gente que habla bobería y dice cosas que no tienen sentido, y atrás viene algún bobo (o boba) y las repite como si fueran ciertas, sin comprobar su veracidad. En este ejercicio se presentan varias de esas “boberías” que la gente dice por ahí. En cada uno de los casos, lo que tienes que hacer es explicar por qué, desde un punto de vista teórico, no tiene sentido lo que se plantea y además comprobar que, a pesar de no tener ningún sentido teórico, en esta cruel, despiadada y anti-teórica realidad nuestra sí se cumple :-/.

- a) (25000 créditos) Cuando se usa la expresión (2) del ejercicio 7 para aproximar $f'_1(1)$ con $h = 30$, se obtiene una mejor aproximación de la derivada que si se usa $h = 0,1$.
- b) (25000 créditos) La función:

$$f(x) = \begin{cases} 0, & 0 \leq x < 1 \\ 1, & x \geq 1 \end{cases},$$

se puede implementar en Python de la siguiente forma:

⁹Aquí la palabra clave es **exacta**. La solución tiene que ser la solución exacta, y no una aproximación calculada por algún método de las bibliotecas *Numpy*, *Scipy* o alguna otra similar. Pista: Este ejercicio pudiera estar en una clase práctica de Álgebra en primer año.

```
def funcion_identidad(x):
    for i in range(70):
        x = math.sqrt(x)
    for i in range(70):
        x = x*x
    return x
```

- c) (25000 créditos) En las computadoras, la suma no es asociativa. Por ejemplo, en dependencia del orden en que se sumen los números:

```
[1e100, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83]
```

se pueden obtener resultados diferentes.

Ejercicios para los muuuuy valientes¹⁰

Voz de tutorial: Por lo general, los ejercicios en esta tercera sección de las clases prácticas son “interesantes”. Para poder resolverlos suele ser usual combinar conocimientos y habilidades que se tienen, con otros conocimientos y habilidades que no se tienen y que hay que buscar y desarrollar para cada ejercicio particular. Hay ocasiones en que la habilidad más útil es la de saber buscar la respuesta, y la segunda más importante, es poder entender lo que se encontró, así que ¡suerte con eso!

Ejercicio 11: Este ejercicio es mucho más difícil de lo que parece. (100000 créditos)

Implemente una función en Python que reciba una lista de números y devuelva la suma. Compruebe su implementación con la lista:

```
[1e100, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83, 1e83],
```

que al sumar sus números se debe obtener: 1.00000000000000001e+100

Ejercicio 12: Y hay quien dice que este es el ejercicio más difícil de todos :-o (Tu propuesta + 100 000 créditos)

Propón un ejercicio relacionado con el tema de la CP, que no esté incluido y que consideres que le aportaría algo “interesante” a la clase ¹¹.

¹⁰O para todos aquellos que estén dispuestos a terminar la asignatura con 5 ;-).

¹¹Voz de tutorial: En este caso, la cantidad de créditos que recibirías es 100 000 + una cantidad que debe depender de la complejidad del ejercicio. Ejercicios sencillos tienen pocos créditos, ejercicios más complejos otorgan más créditos. En esta pregunta, los créditos los ganas solo por proponer el ejercicio. Si además, entregas la solución del ejercicio :-o, ganarías (otra vez) la cantidad de créditos indicados en la orden. Si la cantidad de créditos indicados como parte de la propuesta no está acorde a la complejidad del ejercicio, se considera que está mal y no se gana ningún crédito :-(.

Bibliografía recomendada

Voz de tutorial: Usualmente en esta sección se presentan elementos que pueden ser útiles para resolver los ejercicios propuestos. Sin embargo, si encuentras algún elemento que creas que debería estar listado aquí y no lo está, lo puedes proponer, y se considera como otro ejercicio de la clase práctica. La cantidad de créditos que se pueden recibir por esas adiciones dependen de lo “valioso” que sea el recurso propuesto, pero nunca sería menos de 50000 créditos.

- *Numerical Analysis. 10th Edition.* J. D. Faires, R. L. Burden, y A. M. Burden. Brooks Cole Publishing, 2016.
- *Análisis Numérico. Décima Edición.* J. D. Faires, R. L. Burden, y A. M. Burden. CENGAGE Learning, 2017.
- *Numpy Beginner’s Guide.* 2nd Edition. Ivan Idris. Packt Publishing. 2013.
- *Numpy Cookbook.* Ivan Idris. Packt Publishing. 2012.
- *Matplotlib for Python Developers.* Sandro Tosi. Packt Publishing. 2009.
- *matplotlib Plotting Cookbook.* Alexandre Devert. Packt Publishing. 2014.
- *Accuracy and stability of numerical algorithms.* 2nd Edition. Nicholas J. Higham SIAM. 2002.

Voz de tutorial: Y bueno, aquí termina la orientación de esta primera clase práctica. El resto queda por ti :-). Recuerda que por cada ejercicio de estos (que entregues) puedes ganar la cantidad de créditos indicada en cada caso, y que además, si lo entregas “antes” de la CP, entonces la cantidad de créditos que recibes se duplica ;-).