

Programación

Licenciatura en Física

Dr. Manuel Pulido

¹ FaCENA, Universidad Nacional del Nordeste, Argentina

² IMIT, CONICET, Argentina

<https://pulidom.github.io/>

Horarios de clases

Jueves. 14-18 horas. Por Google meet

Para contactarse con la asignatura, por preguntas, consultas, lo que necesiten pueden escribir a: prog.fis.unne@gmail.com

Página web de la asignatura: <https://pulidom.github.io/programacion/>

En la página van a encontrar las filminas de las clases, las guías y toda otra información que sea de utilidad (bibliografía, instrucciones, etc).

Practicos/preguntas/discusiones: [zulip](#)

Condiciones de aprobación

Para regularizar: Aprobar dos parciales (60%)

Para promocionar: Aprobar los dos parciales con 80% o mas.

Lo único que requiere esta materia para aprobar es tiempo para leer y practicar en una **computadora**.

Material esencial

Debido a la cuarentena es necesario que tengan acceso a una computadora para poder programar. (Generalmente las clases se dan en el laboratorio de informática del Departamento de Física).

Cualquier computadora con acceso a internet, sea PC o laptop, nueva o muy viejita será de utilidad para la asignatura.

Aquellos que no tengan computadora, si pueden pedir una prestada o practicar en alguna computadora de un vecino/amigo/pariente es lo mejor.

En última instancia si lo único que cuentan es el celular, la recomendación sería que intenten conseguir un **teclado bluetooth** para poder “tippear/teclear” los programas.

Cuenta de correo electrónico

- ▶ Deben crear una cuenta en **gmail (google mail)**.
- ▶ El nombre del usuario (username) debe contener el apellido (obligatorio), el nombre y números si es necesario.
- ▶ Idealmente ramon.rodriguez (pero pueden necesitar agregar mas identificatorios si la cuenta ya existe).
- ▶ Una vez creada la cuenta deben escribir un correo electrónico a la dirección de la asignatura: prog.fis.unne@gmail.com
- ▶ Enviar en el correo: Apellido, Nombres, DNI, Nro Libreta Universitaria, Medio disponible (PC/Laptop/Tablet etc). Grupo que forma (3 estudiantes).

Mas detalles en la página web de la asignatura.

Objetivos del curso

- ▶ Aprender a manejar una computadora.
- ▶ Aprender a pensar algorítmicamente.
- ▶ Aprender a programar (introducción).
- ▶ Aprender a leer datos, procesar datos, grabar datos y graficar (introducción).

Temas a cubrir

- ▶ Conocimiento de sistema operativo linux.
- ▶ Manejo por ventanas y por terminal.
- ▶ Conocimiento básico de shell/bash (lenguaje terminal).
- ▶ Programación en python.
- ▶ Conocimiento básico de métodos numéricos (e.g. evaluación de funciones, derivadas e integrales numéricas).

Bibliografía: Para cada tema hay bibliografía específica. Es una materia en la que se incentiva el 'googleo'.

¿Porque es importante la computación en la Lic. en Física?

Porque la computación en la Lic. en Física?

Actualmente, la computadora es una herramienta indispensable para el trabajo en nuestra carrera:

- ▶ Procesamiento de datos (experimentales/teóricos).
- ▶ Análisis y graficación de datos (experimentales/teóricos).
- ▶ Deducciones analíticas (álgebra simbólica).
- ▶ Procesamiento de textos (html, latex, word, etc).
- ▶ Simulaciones numéricas de problemas complejos (sin resolución analítica).
- ▶ Inteligencia artificial/machine learning aplicada a procesos físicos.

Pilares de la Física (y la ciencia en general)

- ▶ Experimentación u observación
- ▶ Teoría (deducciones a partir de primeros principios)
- ▶ Simulaciones

Que es una computadora?

Que es una computadora?

“Máquina” que permite realizar automaticamente una **serie de procesos lógicos y aritméticos** a través de una CPU (unidad de procesamiento central). Además dispone de canales de interacción entrada/salida (input/output) con:

1. Memoria RAM (random access memory). Muy **rápido** acceso pero se pierde al apagar el equipo. (Relativamente pequeña)
2. Disco rígido (o solid state drive SSDs). Acceso de escritura y lectura **lento** el contenido permanece si se apaga el equipo. (Relativamente grande)
3. Dispositivos varios: Monitor, teclado, parlantes, pen drive, etc.

Sistema operativo

- ▶ Es el “software central” de la computadora que se encarga de administrar todos los dispositivos de la computadora.
- ▶ Es el que interactúa con el usuario a través del teclado.
- ▶ Administra la pantalla.
- ▶ Se encarga de la memoria, disco duro, de arrancar todos los servicios cuando prendemos una computadora y cerrar todos los servicios cuando la apagamos.

Existen muy pocos sistemas operativos: windows, MacOS, **linux**, unix.

Regla de oro: Es esencial cuando apagan una computadora que lo hagan a través del SO (del botón NO!) para que este termine y cierre todos los servicios

Linux



El linux fue creado por Linus Torvalds (Finlandes). Basado en Unix pero para PCs de oficina/hogar. Año de lanzamiento 1991.



Richard Stallman (MIT, USA) es uno de los referentes del conjunto de software cercano al S/O conocidos como GNU utilities (+lisp+emacs+etc). Importante referente del software de código abierto.

Que es programar una computadora?

Que involucra programar?

- ▶ Determinar el problema científico. Fenómeno a estudiar (materia, sistemas complejos, biológicos, sociales, etc).
- ▶ Modelado matemático del proceso a estudiar (ecuaciones).
- ▶ Metodología.
 - ▶ 1. Modelado numérico de las ecuaciones.
 - ▶ 2. Algoritmo. Diagrama de flujo (Receta de cocina).
 - ▶ 3. Implementación en lenguaje computacional.
 - ▶ 4. **Debuging. Tests.** Evaluaciones (casos simples).
- ▶ Simulaciones. (compilación y ejecución del programa).
- ▶ Interpretación de la solución.

Algoritmo

¿Qué es una receta de cocina?

Algoritmo

1. Orden secuencial. Las instrucciones se realizan en orden.
2. Preciso y definido. Cada paso o instrucción esta especificada sin ambigüedad.
3. **No se esperan peras de un olmo.** El algoritmo solo resuelve las instrucciones que se le dan (y no posee datos que no han sido asignados). La computadora no hace milagros.
4. Entrada/Salida. Un algoritmo dispone de datos de entrada y de datos de salida (resultados).

Similitud con la receta de cocina.

Instrucciones

Una **instrucción** es una orden precisa que se le da a la computadora para que realice un determinado proceso.

Las computadoras/lenguajes de programación/ reconocen un muy limitado conjunto de instrucciones. Solo se puede utilizar el conjunto de instrucciones predefinido.

Las instrucciones estan definidas por un **lenguaje de programación**. Cada lenguaje de programación tiene su propio set de instrucciones.

Lenguaje máquina y de alto nivel (C, fortran, perl, **python**, basic, etc).

Cada instrucción tiene una precisa sintaxis. Agregar un punto o una falta de este nos dará error.

Concepto de “variable”

En el **disco rígido/SSD** se guarda la información sobre los archivos o los datos que queremos permanezcan cuando apagamos la máquina.

En la **memoria RAM** guardamos la información de las simulaciones, o la ejecución del programa que estamos realizando.

Esta información en la RAM se guarda a través de **variables**.

Las variables tienen un **nombre identificador** y una asignación de **un lote de memoria**.

La primera instrucción que veremos entonces es la asignación de variables

Asignación de variables

```
superficie = 45    # Asigno a una variable un valor  
pi = 3.15          # Asigno a otra variable un valor  
radio = 20         # Asigno a otra variable un valor  
superficie = 50    # Reasigno la variable superficie otro valor  
superficie = pi * radio^2 # Reasigno la variable superficie
```

Una forma alternativa de representar la asignación: `superficie ← 45`

Pueden describir que esta haciendo la computadora en cada línea?

Las operaciones matemáticas tienen definidos ciertos caracteres ASCII: `*`, `/`, `^` etc

Como quedan los registros de memoria al finalizar?

Ejemplo 1

Realizar un algoritmo para calcular el área de un círculo.

$$\text{área} = \pi \times \text{radio}^2$$

¿Cual sería la receta de cocina para calcular el área?

Pseudo-código área de un círculo

1. Ingresar radio del círculo
2. Asignar a la variable radio el valor ingresado
3. Asignar $\pi = 3.141593$
4. Calcular $\text{área} = \pi \times \text{radio}^2$

Entonces un pseudo-código utiliza un lenguaje natural pero estableciendo en forma estricta el orden y cada instrucción requerida.

Diagrama de flujo

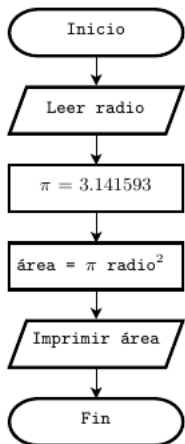


Diagrama de flujo: Es una representación gráfica de un algoritmo.

Cada tipo de instrucción tiene una **figura geométrica**.

Entre instrucciones se ponen flechas/vectores para representar el **sentido del flujo** (orden de ejecución de las instrucciones).

Ejemplo 2

Evaluar y graficar la función cuadrática: $y = ax^2$

Algoritmo evaluación función cuadrática

1. Ingresar el valor de a
2. Ingresar el valor inicial de x , x_0 .
3. Ingresar el valor de la resolución requerida dx .
4. Ingresar número de valores a evaluar n
5. Iniciar un ciclo/loop (de n ciclos)
6. Evaluo la función $y = ax^2$
7. Evaluo nuevo coordenada $x = x_{viejo} + dx$
8. Término el ciclo. Volver al comienzo del ciclo.
9. Impresión/graficación del resultado.

Ejemplo 3

Transformar un número binario a su formato decimal

Algoritmo

1. Ingresar el número $b_n b_{n-1} \cdots b_3 b_2 b_1 b_0$. Como se determinan los dígitos?
2. Asignar $sum = 0$
3. Comienza ciclo i-ésimo desde 0 hasta n .
4. Calcular $r_i = b_i 2^i$
5. Sumar $sum = sum + r_i$
6. Terminar ciclo. Volver al comienzo del ciclo.
7. Imprimir resultado