

19 FEB 2020



Nombre del curso o unidad curricular: Introducción a la Computación

Licenciaturas: Astronomía

Frecuencia y semestre de la formación al que pertenece la unidad curricular: semestre impar, se dicta todo los años

Créditos asignados: 8 - Área Métodos Computacionales

Nombre del/la docente responsable de la unidad curricular y contacto: César Voulgaris
cesarv@fisica.edu.uy

Requisitos previos: se requieren 10 créditos en el área Matemática y 10 en área Física

Ejemplos unidades curriculares de Facultad de Ciencias u otros que aportan dichos conocimientos:

Conocimientos adicionales sugeridos:

Objetivos de la unidad curricular:

a) Herramientas, conceptos y habilidades que se pretenden desarrollar en la unidad curricular

El objetivo principal del curso es introducir al estudiante en la práctica de la programación y desarrollo de algoritmos. Como objetivo secundario, se presentan herramientas específicas de programación necesarias y útiles en las licenciaturas que abarca la materia. La programación es un conocimiento previo importante en muchas de las materias de las licenciaturas, particularmente en los cursos de física computacional, métodos numéricos y laboratorios.



b) En el marco del plan de estudios

En el marco de la formación profesional, ¿qué herramientas aporta esa unidad curricular en la formación profesional de ese estudiante?

Temario sintético de la unidad curricular:

1) Conceptos introductorios de informática

- *Sistemas de numeración**
- *Nomenclatura básica en informática**
- *Arquitectura básica de sistemas informáticos**

2) El sistema operativo (SO) Linux

- *Comandos del SO Linux**
 - Comandos de manejo de archivos
 - Comandos de control del sistema en general
 - Comandos para el acceso a sistemas en red
 - Comandos de filtrado
- *Programando en el SO Linux SHELL Script**

3) Introducción a la Programación

- *Concepto de programa, código fuente y código objeto**
- *Paradigmas de programación**
- *Variables y tipos de datos de un programa**
- *Expresiones, operadores e instrucciones básicas de un programa**
- *Control de flujo de un programa**
 - Condicionales
 - Iteraciones
- *Manejo de entrada y salida y funciones intrínsecas**
- *Procedimientos y funciones**
 - Modularización de un programa: cohesión, independencia y completitud
 - Concepto de reutilización de código
 - Signatura de una función y pasaje de datos
- *Bibliotecas e inclusiones en un programa**
- *Paradigma de Programación Orientada a Objetos (POO)**
 - Encapsulación en objetos en la POO
 - Concepto de clase y herencia en POO
- *Recursividad, concepto y utilidad**

*Tiempo y espacio de ejecución de un programa

*Tópicos avanzados: integración con programas externos y programación paralela



Temario desarrollado:

El punto uno del curso, introduce al estudiante a los conceptos y nomenclatura básica y necesaria en el área de informática. Se introduce el concepto el sistema de numeración en general, con hincapié en el sistema binario y su relación con el decimal. Se introduce los métodos básicos de representaciones numéricas básicas: natural, con signo, entera y de punto flotante.

En el punto dos se introduce el Sistema Operativo (SO) Linux, su origen histórico y utilidad en el ámbito científico. Se recorre los detalles de sus sistema de archivos y los comandos generales (en las distintas funcionalidades básicas que presenta un SO) que permiten controlar al SO. El tópico finaliza con la programación en Linux (Shell Scripting), o sea la integración de los comandos aprendidos en un programa de Linux. Se hace particular hincapié en la utilidad que brinda la programación en Linux, en la accesibilidad de datos y el pre-post procesamiento de los mismos.

En el punto tres, se introduce al estudiante a la programación en general. Se explica los distintos paradigmas de programación, en particular la programación imperativa y estructurada, que son los principales que se usan el curso. Se hace hincapié en cómo funciona un programa, su relación con la arquitectura del computador y el concepto de estado de las variables de un programa.

Se introduce el concepto de variable y tipos de datos y su utilidad. Se enseñan los tipos de datos generales que son universales a todos los lenguajes de programación. Luego las instrucciones básicas del lenguaje de programación en cuestión, el control de flujo del programa y las funciones y procedimientos. A lo largo del aprendizaje se hace hincapié en el concepto de algoritmo y metodologías de desarrollo de un programa con independencia del lenguaje de programación utilizado (o sea de su sintaxis específica). El objetivo principal a lo largo tópico de programación es que el estudiante aprenda cómo construir un programa elaborado en base a criterios, prácticas y metodologías generales de programación: dividir y conquistar, reducción del problema en distintos nivel de abstracción: descripción en lenguaje natural, pseudocódigo de primer y segundo nivel, identificación de partes y su posterior modularización.

Luego de adentrado el curso en la programación estructurada, se introduce el paradigma de Programación Orientada a Objetos (POO), paradigma utilizado en muchos lenguajes actuales de propósito general y en el ámbito académico y científico.

En lo que respecta a las carreras en concreto para las que se dicta el curso, se enseñan a un nivel introductorio básico, nociones de integración en un programa de bibliotecas científicas conocidas, integración con otros lenguajes y nociones de programación en paralelo (como acelerar la ejecución de un programa cuando se dispone de varios procesadores).

Desde su inicio, el curso está orientado al aprendizaje teórico-práctico como una unidad, se introducen los conceptos teóricos, se programan y prueba. Los ejercicios de los prácticos avanzan acorde a la etapa del curso en que se está, desde bloques de programa, algoritmos sencillos a otros más elaborados para luego modularizar y crear un programa más elaborado que se refleja en el nivel de las entregas (ejercicios específicos de los prácticos). Se pretende como finalidad última que el estudiante pueda bajar un problema planteado en lenguaje natural a un programa que lo resuelva e integre aplicaciones o bibliotecas que presenten los resultados pedidos (gráficas, salidas específicas, etc.).



Bibliografía

a) Básica:

a) Linux

Introducción a Linux:

<https://www.tldp.org/LDP/intro-linux/intro-linux.pdf>

Comandos de Linux:

<https://www.tldp.org/LDP/GNU-Linux-Tools-Summary/GNU-Linux-Tools-Summary.pdf>

Programación de Shell:

<https://www.tldp.org/LDP/Bash-Beginners-Guide/Bash-Beginners-Guide.pdf>

UNIX Shell programming (disponible en Biblioteca de Facultad de Ciencias)

b) Programación

Beginning Python: from novice to professional (disponible en Biblioteca de Facultad de Ciencias)

ebook: <https://automatetheboringstuff.com>

ebook: <http://www.packtpub.com/packt/free-ebook/learning-python>

b) Complementaria:

Modalidad cursada: Docencia presencial. El material y la gestión de hitos y entregas es online, vía la página oficial del curso. El dictado de clases es en las aulas de informática, se pretende como criterio un equipo (computadora) por persona.

Metodología de enseñanza:

Carga horaria total: 102

Carga horaria detallada:

a) Horas aula de clases teóricas: 45 teórico-práctico

b) Horas aulas de clases prácticas: 45 teórico-práctico

c) Horas sugeridas de estudio domiciliario durante el período de clase:



Sistema de ganancia de la unidad curricular

Tiene examen final: Si

Se exonera: Si

Nota de exoneración (del 3 al 12): 6

a) Características de las evaluaciones:

***2 parciales escritos de 30 puntos cada uno, que consisten en entregas de programas hechos en el aula.**

***Entregas de ejercicios de 40 puntos en total. Las entregas finales valen más puntos.**

No hay puntaje mínimo por evaluación.

La ganancia del curso se obtiene con 40 puntos en 100 (nota mínima: 3).

La exoneración total del examen final se obtiene con 60 puntos en 100 (nota mínima: 6).

b) Porcentaje de asistencia requerido para aprobar la unidad curricular: 85

c) Puntaje mínimo individual de cada evaluación y total: No hay puntaje mínimo por evaluación. Ganancia de curso se obtiene con 40 puntos en 100

d) Modo de devolución o corrección de pruebas:

Iguá 4225 esq. Mataojo • 11.400 Montevideo – Uruguay
Tel. (598) 2525 0378 • (598) 2522 947 • (598) 2525 8618 al 23 ext. 7 110 y 7 168 • Fax (598) 2525 8617

