

Planificación de la Asignatura: Informática Básica - TUPED

Fecha: 23/10/2024 13:02

Código: T1628

Carrera: Tecnicatura Universitaria en Procesamiento y Explotación de Datos

Departamento Académico: Informática

Docente a cargo:

Correo del docente a cargo: sin datos

Régimen de Dictado: Cuatrimestral 1º Cuatrimestre

Carga Horaria Semanal: 5 horas semanales

Carga Horaria Total: 70 horas

Contenidos Mínimos:

Interacción con sistemas operativos. Elementos y características de los lenguajes de programación.

Entornos de programación. Tipos de datos, variables y constantes. Algoritmos computacionales. Estructuras de control. Estructuras de datos: arreglos y listas. Funciones. Manipulación de archivos.

Correlativas Regulares para cursar:

No posee

Correlativas Aprobadas para cursar:

No posee

Correlativas Aprobadas para promocionar o rendir el examen final:

No posee

Objetivo General:

Que los estudiantes logren:

- Integrar el recurso informático al proceso de formación de fundamento, científica, técnica y tecnológica.
- Comprender y utilizar con criterio los conceptos fundamentales de diseño de algoritmos computacionales y su programación en computadoras digitales.
- Resolver problemas a través de programas de computadora, empleando un lenguaje de programación interpretado de alto nivel.
- Desarrollar el pensamiento lógico y crítico dentro de un contexto de trabajo colaborativo.
- Alcanzar un nivel de pensamiento que le permita extender autodidácticamente sus conocimientos en el área abarcada por la asignatura.

Objetivos Particulares:

Al finalizar el cursado y aprobar la asignatura Informática Básica, el estudiante será capaz de:

- Dominar el concepto de algoritmo computacional y su importancia para resolver problemas de programación utilizando computadoras digitales modernas.
- Diseñar y desarrollar algoritmos computacionales y sus correspondientes programas para resolver distintos problemas de complejidad baja a media.
- Dominar y aplicar los conceptos de programación para el diseño, edición, interpretación, depuración y ejecución de programas.
- Adquirir la destreza necesaria para seleccionar tipos de datos adecuados para organizar apropiadamente soluciones a problemas de baja complejidad.
- Emplear un entorno de desarrollo integrados (IDE, en inglés) utilizando sus principales características de soporte al desarrollo de programas informáticos.
- Reconocer y reutilizar las principales bibliotecas y módulos de código en un lenguaje interpretado de alto nivel.
- Comprender y aplicar criteriosamente las estructuras de datos conocidas como arreglos y listas a la resolución de problemas de programación tomando en cuenta sus principales características.
- Comprender, crear y aplicar estructuras de datos básicas combinadas: registros y listas.
- Comprender y aplicar estructuras de control en un lenguaje de programación de alto nivel.
- Comprender y programar funciones en un lenguaje de programación de alto nivel.
- Diseñar y desarrollar programas de computadora en un lenguaje interpretado de alto nivel para manipular archivos digitales en dispositivos de almacenamiento secundario para creación, acceso,

modificación, movimiento, borrado y verificaciones.

• Aplicar criterios de buenas prácticas de programación.

• Trabajar colaborativamente para resolver problemas de programación.

Programa Analítico:**Unidad 1: Introducción a la Programación**

Concepto de algoritmo y algoritmos computacionales. Programa. Depuración y documentación de programas. Lenguajes de Programación: tipos y características. Paradigmas de programación.

Unidad 2: Introducción al Lenguaje Python

Estructura de un programa en Python y su ejecución por parte del Intérprete. Tokens del lenguaje. Variables. Ámbito de validez de las variables. Flujos de entrada y salida.

Unidad 3: Expresiones y Lógica de Control

Operadores en Python. Jerarquía de los operadores. Construcción de expresiones. Estructuras de control: condicional (if, elif, else) iterativas (while, for). Uso de las declaraciones pass, break, continue. Función range(). Módulo random. Resolución de problemas mediante programas en Python. Edición, prueba y depuración de programas empleando un Intérprete de Python.

Unidad 4: Estructuras de datos

Listas: creación, recorrido, añadir y quitar elementos, métodos integrados. Listas por comprensión. Tuplas. Problemas de programación empleando datos estructurados.

Unidad 5: Funciones y módulos

Diseño e implementación de funciones en Python. Argumentos, argumentos por defecto, argumentos arbitrarios. Funciones integradas en Python. Importación de módulos y paquetes. Modularidad: creación de módulos con funciones. Problemas de programación empleando funciones. Recursividad

Unidad 6: Módulo String

Definición y creación, eliminación, modificación, recorrido de objetos tipo String. Métodos integrados. Operadores y operaciones. Problemas de programación empleando cadenas de caracteres.

Unidad 7: Archivos de texto

Apertura. Lectura. Escritura. Cierre. Funciones integradas vinculadas al manejo de archivos. Archivos de texto estructurados: CSV

Listado de Actividades de Formación Práctica:

Introducción a Python, variables y operadores

Estructuras de control de flujo

Estructuras de datos

Funciones en Python

Cadenas en Python

Entrada y Salida: archivos de texto

Metodología de Evaluación Durante el cursado:**Evaluaciones parciales**

Se realizarán 2 evaluaciones parciales de 2 hs de duración. En la primera se evaluarán los temas hasta la unidad 3 de la asignatura. Al final del cuatrimestre se tomará un examen integrador final de la totalidad de los temas dictados.

Ambas instancias de evaluación se pueden recuperar.

La nota para aprobar las evaluaciones es 6 o superior.

Metodología de Evaluación en Exámenes Finales:**Examen final para alumnos regulares:**

Se realizará una evaluación de 2 o 3 problemas a resolver mediante código en Python y preguntas conceptuales. Tiempo estimado de resolución: 2 Hs. Calificación mínima para la aprobación: 6.

Examen final para alumnos libres:

El alumno en condición de libre debe presentar la resolución de un trabajo práctico. Se brindarán con antelación el enunciado y los archivos para procesar. El día de la mesa, el estudiante deberá presentar un código funcional, que resuelva las actividades planteadas, las cuales se evaluarán mediante un cuestionario. Una vez aprobada esta instancia, pasará a rendir el mismo examen que el alumno regular. Calificación mínima para la aprobación: 6 en ambas instancias (examen escrito y trabajo práctico).

Condiciones de Regularidad :

Condiciones de Regularidad

Para poder regularizar se exige que el alumno apruebe el primer examen parcial y el examen integrador final con nota igual o superior a 60 puntos y tenga un 75% de asistencia a las clases de práctica.

Condiciones de promoción

La asignatura puede ser aprobada (promovida) durante el cursado, evitando la instancia del examen final.

La condición para promover es tener 75% de asistencia en la práctica y nota igual o superior a 75 puntos en el examen integrador final.

Bibliografía Principal:

El tutorial de Python. Python Software Foundation. Disponible online en: <https://docs.python.org/es/3/tutorial/>

La Biblioteca Estándar de Python. Python Software Foundation. Disponible online en:

<https://docs.python.org/es/3/library/index.html#library-index>

Referencia del Lenguaje Python. Python Software Foundation. Disponible online en:

<https://docs.python.org/es/3/reference/index.html#reference-index>

Bibliografía Complementaria:

Tutorial de Python: Guía básica de Python en español. Juan José Lozano Gómez. Disponible en:

<https://j2logo.com/python/tutorial/>

Programación en Python - Nivel básico. Repositorio de manuales y recursos del entrenamiento realizado por la empresa Covantec R.L. Disponible en:

<https://entrenamiento-python-basico.readthedocs.io/es/latest/index.html>.

Introducción a la programación con Python. Apuntes del Curso de iniciación a la programación en Python, del módulo Lenguaje de Marcas y Sistemas de Gestión de la Información del ciclo formativo Administración de Sistemas Informáticos en Red (ASIR), IES Abastos de Valencia (España), 2020/2021. Disponible en:

<https://www.mclibre.org/consultar/python/>.

Python Tutorial. Bernd Klein. Disponible en: <https://python-course.eu/python-tutorial/>

Lógica de Programación. Fundación Carlos Slim. Disponible en:

https://aprende.org/pages.php?r=cfc_course&tagID=7929