



UTEC
Universidad
de Ingeniería
y Tecnología

CIENCIA DE LA
COMPUTACIÓN

UTEC
UNIVERSIDAD DE INGENIERIA
Y TECNOLOGIA



ÍNDICE

ASIGNATURA	3
DATOS GENERALES	3
Créditos: cuatro (4) créditos	3
Horas de teoría: dos (2) semanales	3
Horas de práctica: cuatro (4) semanales	3
Duración del período: dieciséis (16) semanas	3
Condición: Obligatorio	3
Modalidad: Presencial	3
Requisitos: Ninguno	3
PROFESORES	3
Profesor coordinador del curso	3
Profesor(es) instructor(es) del curso	3
INTRODUCCIÓN AL CURSO	4
OBJETIVOS	4
COMPETENCIAS	5
RESULTADOS DE APRENDIZAJE	5
TEMAS	6
PLAN DE TRABAJO	7
Metodología	7
Sesiones de teoría	7
Sesiones de práctica (laboratorio o taller)	7
Exposiciones individuales o grupales:	7
SISTEMA DE EVALUACIÓN	8
REFERENCIAS BIBLIOGRÁFICAS	8

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

SILABO 2020-2

1. ASIGNATURA

CS1100- Introducción a Ciencia de la Computación

2. DATOS GENERALES

2.1 Créditos: cuatro (4) créditos

2.2 Horas de teoría: dos (2) semanales

2.3 Horas de práctica: cuatro (4) semanales

2.4 Duración del período: dieciséis (16) semanas

2.5 Condición: Obligatorio

2.6 Modalidad: Virtual

2.7 Requisitos: Ninguno

3. PROFESORES

3.1 Profesor coordinador del curso

Jesus Edwin Bellido Angulo, (jbellido@utec.edu.pe)

Horario de atención: Lunes de 17:00 a 18:00 horas.

Alan Victor Raul Morante Ponce, (amorantep@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

3.2 Profesor(es) instructor(es) del curso

Jesus Edwin Bellido Angulo, (jbellido@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Maria Hilda Bermejo Rios, (mbermejo@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Jaime Moshe Farfán Madariaga, (jfarfan@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Henry Giovanni Gallegos Velgara, (hgallegos@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Alan Victor Raul Morante Ponce, (amorantep@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Patricio Morriberón Cornejo, (pmorriberon@utec.edu.pe)

Horario de atención: Previa coordinación con el profesor

Fecha de actualización: 27/08/2020

Revisado y aprobado por el Centro de Excelencia en Enseñanza y Aprendizaje y la Dirección de Ciencia de la Computación



Fernando Augusto Nuñez Calderon, (fnunez@utec.edu.pe)
Horario de atención: Previa coordinación con el profesor

4. INTRODUCCIÓN AL CURSO

Este es el primer curso en la secuencia de los cursos introductorios a la Ciencia de la Computación. En este curso se pretende cubrir los conceptos señalados por la Computing Curricula IEEE-CS/ACM 2013. La Programación es uno de los pilares de la Ciencia de la Computación; cualquier profesional del área necesitará programar para concretar sus modelos y propuestas. Este curso introduce los conceptos fundamentales de este arte. Los tópicos incluyen tipos de datos, estructuras de control, funciones, listas, recursividad y la mecánica de la ejecución, prueba y depuración.

5. OBJETIVOS

Sesión 1: Discutir la importancia de los algoritmos en el proceso de solución de un problema.

Sesión 2: Identificar, describir y escribir programas que usan tipos de datos primitivos

Sesión 3: Escoger estructuras de condición adecuadas para una tarea de programación dada.

Sesión 4: Escoger estructuras de repetición adecuadas para una tarea de programación dada.

Sesión 5: Implementar un algoritmo de divide y vencerás para resolver un problema usando funciones.

Sesión 6: Identificar y describir el uso de cadenas de texto y escribe programas que usan cadenas de texto.

Sesión 7: Analizar y explicar el comportamiento de programas simples que involucran estructuras fundamentales de programación, variables, expresiones, asignaciones, E/S, estructuras de control, funciones, paso de parámetros.

Sesión 8: Escribir un programa que usa E/S de archivos para brindar persistencia a través de ejecuciones múltiples

Sesión 9: Describir el concepto de recursividad y da ejemplos de su uso. Identifica el caso base y el caso general de un problema basado en recursividad y escribe programas basado en funciones recursivas.

Sesión 10: Diseñar, implementar, probar, y depurar un programa que usa estructuras de datos como arreglos, listas y tuplas.

Sesión 11: Diseñar, implementar, probar, y depurar un programa que usa estructuras de datos como diccionarios o tablas de hash.

Sesión 12: Usar la notación formal Big O para dar límites de casos esperados en el tiempo de complejidad de los algoritmos. Explica a que se refiere con “mejor”, “esperado” y “peor” caso de comportamiento de un algoritmo

Sesión 13: Implementar algoritmos de ordenamiento y explicar las diferencias en sus tiempos de complejidad.

Sesión 14: Implementar algoritmos de búsqueda simple y explicar las diferencias en sus tiempos de complejidad.

6. COMPETENCIAS

Competencias para Ciencia de la Computación:

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina (*nivel 2*)
El estudiante aplica el pensamiento computacional para resolver problemas de manera automática.

- b) Analizar problemas, identificar y definir los requerimientos computacionales apropiados para su solución (*nivel 2*)
El estudiante analiza el costo computacional requerido de un algoritmo para dar solución a un problema.

Competencias para Ingeniería:

- a3) Capacidad de aplicar conocimientos de ingeniería (**nivel 2**)
El estudiante aplica el pensamiento computacional para resolver problemas de manera automática.

- k1) Capacidad de utilizar las técnicas, las habilidades y las herramientas de la ingeniería moderna necesarias para la práctica de la ingeniería (**nivel 2**).
El estudiante analiza el costo computacional requerido de un algoritmo para dar solución a un problema.

7. RESULTADOS DE APRENDIZAJE

Al finalizar el curso de Introducción a la ciencia de la computación se espera:

RA1. Que el estudiante sea capaz de aplicar conocimientos de computación y matemática demostrando dominio del tema, orden, coherencia en los mismos.

RA2. Que el estudiante sea capaz de analizar problemas e identificar soluciones algorítmicas, haciendo uso correcto de lenguajes de programación.

RA3. Que el estudiante sea capaz de desarrollar capacidades de abstracción, encontrando patrones que puedan ayudar a encontrar la solución de problemas y cómo resolverlos.

8. TEMAS

1. Introducción al curso

2. Algoritmos

- 2.1. ¿Qué es un algoritmo?
- 2.2. ¿Cómo se representa?

3. Introducción a Python

- 3.1. ¿Qué es Python?
- 3.2. ¿Qué es un intérprete?
- 3.3. ¿Qué es un IDE?
- 3.4. Entradas y Salidas
- 3.5. Tipos de datos básicos
- 3.6. Variables
- 3.7. Operadores aritméticos y lógicos

4. Estructuras de Control

- 4.1. Estructuras de control selectivas
- 4.2. Estructuras de control repetitivas
- 4.3. Estructuras de control repetitivas anidadas

5. Strings

- 5.1. ¿Qué es un *string*?
- 5.2. Funciones básicas de strings
- 5.3. Índices y operadores de strings
- 5.4. Iteración en un string

6. Funciones

- 6.1. ¿Qué es una función?
- 6.2. Parámetros por valor y referencia
- 6.3. Librerías
- 6.4. Alcance de una variable
- 6.5. Parámetros por defecto

7. Listas

- 7.1. ¿Qué es una lista?
- 7.2. Índices y operadores en una lista
- 7.3. Iteración en una lista
- 7.4. Matrices
- 7.5. Listas por comprensión

8. Diccionarios

- 8.1. ¿Qué es un diccionario?
- 8.2. Dupla: Llave - Valor

9. Archivos

- 9.1. Lectura de archivos
- 9.2. Escritura de archivos

10. Complejidad algorítmica

- 10.1. Complejidad en tiempo

10.2. Complejidad en espacio

11. Recursión

11.1. ¿Qué es recursión?

11.2. Cola de llamadas

11.3. Backtracking

12. Ordenamiento

12.1. Ordenamiento por selección

12.2. Ordenamiento por inserción

12.3. Ordenamiento rápido (*quicksort*)

13. Búsqueda Binaria

9. PLAN DE TRABAJO

9.1 Metodología

El curso presenta una metodología de aprendizaje activo, basada en la resolución de problemas de una manera práctica con enfoque al conocimiento adquirido en sesiones teóricas. Adicionalmente, existe aprendizaje a través de la ejecución de un proyecto, de esta manera, se anima y motiva al estudiante a que continúe con su proceso de aprendizaje.

9.2 Sesiones de teoría

Las sesiones de teoría se llevan a cabo en clases magistrales donde se realizarán actividades que propicien un aprendizaje activo, con dinámicas que permitan a los estudiantes interiorizar los conceptos

9.3 Sesiones de práctica (laboratorio o taller)

Para verificar que los alumnos hayan alcanzado el logro planteado para cada una de las unidades de aprendizaje, realizarán actividades que les permita aplicar los conocimientos adquiridos durante las sesiones de teoría y se les pondrá retos que permitan evaluar el desempeño de los alumnos.

10. SISTEMA DE EVALUACIÓN

	TEORÍA	PRÁCTICA Y/O LABORATORIO
EVALUACIÓN *La ponderación de la evaluación se hará si ambas partes están aprobadas	1 Examen calificado - E1 (40%)	2 evaluaciones continuas - C1 y C2 (4% cada una) 4 Prácticas calificadas - PC1, PC2, PC3 y PC4 (9% cada una) 2 evaluaciones de Proyectos - P1 (8%) y P2 (8%)
	$\square = \frac{\square I}{40}$	$\square = \frac{4(\square I + \square 2) + 9(\square \square 1 + \square \square 2 + \square \square 3 + \dots)}{60}$
	40%	60%
	100%	

Las rúbricas que permitirán medir las actividades más significativas del curso y que, además se relacionan con la evaluación de las competencias del estudiante son: [enlace](#)

11. SESIONES DE APOYO O TUTORIAS

Este apartado permite formalizar los espacios de apoyo a los estudiantes y que éstos tengan la atención NECESARIA y el tiempo disponible para presentar sus dudas y consultas acerca del curso:

Semana	Fecha/ Hora	Tema tratar	Objetivos de la sesión
2	11/09/2020 10:00 AM – 11:00 AM	Configuración	Revisar la instalación y solucionar los problemas presentados.
4	25/09/2020 10:00 AM – 11:00 AM	Tarea 1	Resolver las dudas del enunciado de la Tarea 1
6	09/10/2020 10:00 AM – 11:00 AM	Práctica Calificada 1	Solución de la Práctica Calificada 1
8	23/10/2020 10:00 AM – 11:00 AM	Práctica Calificada 2	Solución de la Práctica Calificada 2
10	06/11/2020	Práctica	Solución de la Práctica

Fecha de actualización: 27/08/2020

Revisado y aprobado por el Centro de Excelencia en Enseñanza y Aprendizaje y la Dirección de Ciencia de la Computación

	10:00 AM – 11:00 AM	Calificada 3	Calificada 3
12	20/11/2020 10:00 AM – 11:00 AM	Práctica Calificada 4	Solución de la Práctica Calificada 4
14	04/12/2020 10:00 AM – 11:00 AM	Proyecto Final	Resolver dudas del proyecto Final
16	18/12/2020 10:00 AM – 11:00 AM	Examen	Solución del Examen Final.

11. REFERENCIAS BIBLIOGRÁFICAS

1. Brookshear, J. G., Smith, D., & Brylow, D. (2012). Computer Science: An Overview.
2. Guttag, J. (2016). Introduction to computation and programming using Python: With application to understanding data. MIT Press.
3. Stephenson, B. (2016). The Python Workbook. SPRINGER INTERNATIONAL PU.