



MODELO DE SÍLABO ADAPTADO PARA EL PERIODO DE ADECUACIÓN A LA EDUCACIÓN NO PRESENCIAL

Facultad de Ingeniería
Escuela Profesional de Ingeniería Industrial

SÍLABO 2021-II

I. DATOS ADMINISTRATIVOS

1. Asignatura	: PROGRAMACIÓN DE COMPUTADORAS.
2. Código	: ACP 001.
3. Naturaleza	: Teórico-práctico.
4. Condición	: Obligatorio.
5. Requisitos	: Ninguno
6. Nro. Créditos	: 3.5
7. Nro de horas	: 2 Teóricas / 3 Laboratorio.
8. Semestre Académico	: 2.
9. Docente	: Mg. Ada Cebreros Delgado de la Flor.
10. Correo Institucional	

II. SUMILLA

Propósitos generales:

Tiene como propósito brindar al estudiante los fundamentos de la computación y la programación. Identificar los conceptos básicos de diseño de algoritmos tales como la representación de los datos, estructuras, bucles, condicionales, arreglos, así como la utilidad de un buen diseño algorítmico y modular: funciones. Identifica la estructura general de un programa en lenguaje estructurado. Aplica una metodología de programación para una adecuada resolución de problemas.

Síntesis del contenido:

Conceptos básicos de computación y programación, estructura básica de un programa; el lenguaje de programación C: operadores, expresiones, control del flujo de un programa, funciones; manejo de bits; el lenguaje de programación C: arreglos y cadenas de caracteres; Interacción con dispositivos electrónicos programables.

III. COMPETENCIAS GENÉRICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Pensamiento crítico y creativo
- Resolución de Problemas.

IV. COMPETENCIAS ESPECÍFICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Diseño en Ingeniería
- Solución de Problemas de Ingeniería
- Gestión de Proyectos.

V. DESARROLLA EL COMPONENTE DE: INVESTIGACIÓN () RESPONSABILIDAD SOCIAL (x)

VI. LOGRO DE LA ASIGNATURA

Al finalizar la asignatura el estudiante:

- Identifica los datos, procesos y resultados que deben ser considerados en un programa y plantea la solución mediante algoritmos y diagramas de flujo.
- Codifica instrucciones de entrada / salida para la lectura de datos y escritura de resultados. Desarrolla programas utilizando operaciones y funciones matemáticas
- Programa instrucciones de decisión con if-else e instrucciones de repetición con while y do-while. Desarrolla programas utilizando decisiones y repeticiones



Universidad Ricardo Palma
Rectorado
Oficina de Desarrollo Académico, Calidad y Acreditación

- Programa instrucciones de control con for e instrucciones de selección con switch case. Desarrolla programas utilizando control de loops y selección de bloques para distintos casos.
- Desarrolla programas con arreglos de una dimensión para el uso de vectores y de cadenas. Desarrolla programas de dos dimensiones para el uso y operaciones con matrices.

VII. PROGRAMACIÓN DE CONTENIDOS

UNIDAD 1: FUNDAMENTOS DE PROGRAMACIÓN	
LOGRO DE APRENDIZAJE: Identifica los datos, procesos y resultados que deben ser considerados en un programa y plantea la solución mediante algoritmos y diagramas de flujo.	
Semana	Contenido
1	Introducción. Concepto de programa e instrucciones. Lenguajes de programación y Códigos fuentes. Definición de problemas y Casuísticas. Concepto de programa e instrucciones. Pasos de la programación.
2	Concepto y características de los Algoritmos. Algoritmos de decisiones, repeticiones y de control. Diagramas de flujo y sus Símbolos. Diagramación y Pseudocódigos. Diagramas con decisiones, repeticiones, Control y selección.

UNIDAD 2: INSTRUCCIONES DE ENTRADA / SALIDA Y OPERACIONES	
LOGRO DE APRENDIZAJE: Codifica instrucciones de entrada / salida para la lectura de datos y escritura de resultados. Desarrolla programas utilizando operaciones y funciones matemáticas.	
Semana	Contenido
3	La estructura de un programa, clases y métodos. Reglas de la programación y sus Librerías Concepto de datos, constantes y variables. Tipos de datos. Declaración de variables: int, float, char, long, double. Instrucciones de entrada Console.ReadLine() Instrucciones de salida Console.Write()
4	Operadores aritméticos y sus reglas. Niveles de datos. Operaciones matemáticas, prioridades y uso de los paréntesis. Instrucciones de asignación y sus reglas. Conversiones de tipo. Funciones Math.: Abs(), Sqrt(), Sin(), Cos(), Tan(), Log(), Log10(), Pow(), Exp(), Round(), PI. Asignaciones múltiples y operador incluido. Operadores: ++, --.
5	Monitoreo y Retroalimentación. Evaluación del Logro

UNIDAD 3: INSTRUCCIONES DE DECISION Y DE REPETICION	
LOGRO DE APRENDIZAJE: Programa instrucciones de decisión con if-else e instrucciones de repetición con while y do-while. Desarrolla programas utilizando decisiones y repeticiones.	
Semana	Contenido
6	Operadores relacionales: ==, !=, >, <, >=, <=. Operadores lógicos: !, &&, , Reglas lógicas. Algoritmos y diagramas con decisiones Lógicas. Instrucción if-else, sintaxis y reglas. Instrucción sólo if, if-else con bloques, If, dentro de if. Algoritmos y programas con if-else.
7	Instrucción while, sintaxis y reglas. Instrucciones while con una instrucción y con bloque de instrucciones. Algoritmos y programas con while. Instrucción do-while, sintaxis y reglas. Instrucciones do while con una instrucción y con bloque de instrucciones Algoritmos y programas con do-while.
8	Algoritmos y programas combinados usando If-else, while, do while, y funciones matemáticas
9	Monitoreo y Retroalimentación. Evaluación del Logro

UNIDAD 4: INSTRUCCIONES DE CONTROL Y DE SELECCIÓN	
LOGRO DE APRENDIZAJE: Programa instrucciones de control con for e instrucciones de selección con switch case. Desarrolla programas utilizando control de loops y selección de bloques para distintos casos.	
Semana	Contenido



10	Instrucción switch-case, sintaxis y reglas. Uso del default en el switch-case. Selección con switch-case y default. Algoritmos y programas con switch-case. Instrucción break, sintaxis y reglas. Uso del break dentro del switch-case.
11	Instrucción for, expresiones de inicio, Evaluación y preparación. Control, sintaxis y reglas. Instrucción for con una instrucción y con bloque de instrucciones. Instrucciones for anidadas, for "interior" y for "exterior". Algoritmos y programas con for y Con for anidado "nesting".
12	Algoritmos y programas combinados, usando for , If-else, while, do while, switch
13	Monitoreo y Retroalimentación. Evaluación del Logro

UNIDAD 5: ARREGLOS CADENAS Y MATRICES

LOGRO DE APRENDIZAJE: Desarrolla programas con arreglos de una dimensión para el uso de vectores y de cadenas. Desarrolla programas de dos dimensiones para el uso y operaciones con matrices.

Semana	Contenido
14	Concepto de arreglo, declaración, subíndices. Sintaxis y reglas. Lectura y escritura de arreglos de una dimensión. Ordenamiento y eliminación de elementos de un arreglo. Operaciones con arreglos unidimensionales. Arreglos de cadenas "string", Lectura y escritura de cadenas. Comparaciones e inicializaciones de Cadenas.
15	Arreglos bidimensionales, sintaxis y reglas. Subíndices, direcciones y declaración de arreglos bidimensionales. Lectura, escritura y operaciones con arreglos bidimensionales. Arreglos multidimensionales, sintaxis, reglas y usos. Programas con matrices: suma, multiplicación.
16	Monitoreo y Retroalimentación.
17	Examen sustitutorio

VIII. ESTRATEGIAS DIDÁCTICAS

Aula invertida, Aprendizaje Colaborativo, Disertación.

IX. MOMENTOS DE LA SESIÓN DE APRENDIZAJE VIRTUAL

La modalidad no presencial desarrollará actividades sincrónicas (que los estudiantes realizarán al mismo tiempo con el docente) y asincrónicas (que los estudiantes realizarán independientemente fortaleciendo su aprendizaje autónomo. La metodología del aula invertida organizará las actividades de la siguiente manera:

Antes de la sesión

Exploración: preguntas de reflexión vinculada con el contexto, otros.

Problematización: conflicto cognitivo de la unidad, otros.

Durante la sesión

Motivación: bienvenida y presentación del curso, otros.

Presentación: PPT en forma colaborativa, otros.

Práctica: resolución individual de un problema, resolución colectiva de un problema, otros.

Después de la sesión

Evaluación de la unidad: presentación del producto.

Extensión / Transferencia: presentación en digital de la resolución individual de un problema.

IX. EVALUACIÓN

La modalidad no presencial se evaluará a través de productos que el estudiante presentará al final de cada unidad. Los productos son las evidencias del logro de los aprendizajes y serán evaluados a través de rúbricas cuyo objetivo es calificar el desempeño de los estudiantes de manera objetiva y precisa.

Retroalimentación. En esta modalidad no presencial, la retroalimentación se convierte en aspecto primordial para el logro de aprendizaje. El docente devolverá los productos de la unidad revisados y realizará la retroalimentación respectiva.

UNIDAD	INSTRUMENTOS	PORCENTAJE
I	Rúbrica	25%
II	Rúbrica	25%
III	Rúbrica	25%
IV	Rúbrica	25%



La evaluación sustitutoria 5, reemplaza la nota más baja de las evaluaciones 1 a 4.

$$\text{Nota Final} = \frac{(\text{Ev1} + \text{Ev2} + \text{Ev3} + \text{Ev4})}{4}$$

X. RECURSOS

- Equipos: computadora, laptop, Tablet, celular
- Materiales: apuntes de clase del Docente, separatas de problemas, lecturas, videos.

XI. REFERENCIAS

Bibliografía Básica.

- 1 H,M.Deitel / P.J.Deitel: "Como programar en C#. 3ra.Ed. Prentice-Hall, 2013.
2. Ceballos F. "Micorosoft C# Curso de programación" Ed. RA-MA, España, 2011
3. Cebreros Ada. "Introducción al C#", Programación de Computadoras – Separata Teoría y Práctica
4. Gary Bronson, "C++ para Ciencias e ingeniería". 2da Edic. 2015
5. Joyanes L. "Programación C#" Ed. Mc.Graw Hill, España, 2001.
6. Maynard Kong. "Lenguaje de Programación C" Ed. Pontificia Universidad Católica del Perú, 1994.
7. Joyanes L. "Fundamentos de Programación" Ed. Mc.Graw Hill, 1993.
8. Farina Mario: "Diagramas de Flujo" Ed. Diana-México, 1992.
9. Lozano Letvin: "Diagramación y Programación" Ed. Mc.Graw Hill, 1991.
10. Brian W. Kernighan & Dennis M. Ritchie: "Lenguaje de Programación C" Ed. Prentice-Hall, 1991.