



UNIVERSIDAD RICARDO PALMA
FACULTAD DE INGENIERÍA
ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INDUSTRIAL

PLAN DE ESTUDIOS 2015-II

I. DATOS GENERALES

Asignatura	: PROGRAMACIÓN DE COMPUTADORAS
Código	: ACP-001
Área Académica	: Sistemas
Condición	: Obligatorio
Nivel	: 2do.
Créditos	: 3.5
Horas de Teoría	: 2
Horas de Laboratorio	: 3
Requisitos	: Ninguno
Profesor	: Mg. Ada Cebreros Delgado de la Flor.

II. SUMILLA

La asignatura tiene un contenido teórico-práctico complementado con laboratorio. Tiene como propósito capacitar al alumno en el aprovechamiento de los lenguajes de programación para ser aplicados en sus cursos posteriores y en su carrera profesional. Se imparte contenidos de programación, empleando conceptos previamente aprendidos por el estudiante tales como sistemas operativos Windows e Internet con aula virtual en Intranet. El curso desarrolla temas tales como: diseño de algoritmos, funciones estándar, instrucciones de decisión, repetición, selección y arreglos.

III. COMPETENCIAS GENERICAS A LAS CONTRIBUYE LA ASIGNATURA DE LA CARRERA

1. Pensamiento crítico y creativo
2. Resolución de problemas

IV. COMPETENCIAS ESPECÍFICAS A LAS QUE CONTRIBUYE LA CARRERA

1. Diseño en Ingeniería
2. Solución de Problemas de Ingeniería
3. Gestión de Proyectos

V. LOGROS DE LA ASIGNATURA

Al finalizar el estudiante

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

4. 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.
5. 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.

VI. 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 anidadadas, 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

VII. ESTRATEGIAS DIDÁCTICAS

Disertación, Aprendizaje Basado en Proyectos, Problemas, Juegos; Aprendizaje Colaborativo, Aprendizaje Basado en Investigación, Estudio de Casos, Talleres, etc.

Se podrán desarrollar actividades sincrónicas (que los estudiantes realizarán al mismo tiempo con el docente) y asincrónicas (que los estudiantes realizarán independientemente fortaleciendo su aprendizaje autónomo).

La planificación y ejecución de las sesiones de aprendizaje deberán considerar actividades que se organizarán de la siguiente manera:

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

Problematización: conflicto cognitivo de la unidad, otros.

Motivación: bienvenida y presentación del curso, otros.

Presentación: PPT, otros.

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

Evaluación de la unidad: presentación del resultado o producto.

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

IX. EVALUACIÓN

Las evaluaciones se realizarán a lo largo del semestre con el propósito de determinar en qué medida el estudiante va logrando las competencias de la asignatura.

Las actividades de enseñanza se complementarán con actividades de evaluación continua (AEC) tales como: laboratorios, talleres, proyectos, trabajos, simulaciones, exposiciones, controles de lectura, casos, participaciones en las sesiones de clases, entre otras, para las cuales se podrán seleccionar los instrumentos que el docente estime conveniente, además cuando menos de una rúbrica como recurso educativo.

Los exámenes parcial y final se realizarán en las semanas 8 y 16.

El promedio final de la asignatura se obtendrá de la manera siguiente:

Laboratorios Calificados	: L1, L2, L3 y L4	$PL = \frac{L1 + L2 + L3 + L4}{4}$
Examen Final	: EF	
Examen Parcial	: EP	$PF = \frac{EP + EF + PL}{3}$
Examen Sustitutorio (**)	: ES	
Promedio Final	: PF	

(**) El Examen Sustitutorio reemplaza la nota más baja de los exámenes y se realizará en la semana 17.

X. RECURSOS

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

XI. RECURSOS BIBLIOGRÁFICOS

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

BASE DE DATOS URP:

Diseño y Construcción de algoritmos– teoría y práctica - 2015

Autor: Mancilla Herrera – Universidad del Norte.

<https://elibro.net/es/lc/bibliourp/titulos/69931>

Diseño Industrial por Computador

Editorial: Marcombo -1987

<https://elibro.net/es/lc/bibliourp/101859>

