

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

ESTUDIOS GENERALES CIENCIAS

PROGRAMA ANALÍTICO

CURSO	: FUNDAMENTOS DE PROGRAMACIÓN (FPRO)
CLAVE	: 1INF01
TIPO	: OBLIGATORIO PARA TODAS LAS ESPECIALIDADES EXCEPTO QUÍMICA E INGENIERÍA INDUSTRIAL
CRÉDITOS	: 3.00
HORAS DE:	
TEORÍA	: 5 SEMANALES
LABORATORIO	: 4 SEMANALES
REQUISITOS	: (1FIS02)
CICLO DE VERANO	: 2024

I. Objetivos del curso

Al terminar el semestre el alumno comprenderá y aplicará la lógica de proposiciones empleando la negación, la conjunción y la disyunción en la representación del conocimiento. Reconocerá lo que se entiende por algoritmo identificando las principales características del mismo. Comprenderá y aplicará los pasos clásicos para la resolución de problemas usando el computador, así como el funcionamiento de las estructuras selectivas e iterativas. Será capaz de explicar cómo funcionan los algoritmos expresados en diagrama de flujo y podrá diseñar algoritmos expresados en pseudocódigos. Implementará algoritmos que contengan estructuras selectivas e iterativas usando un lenguaje de programación imperativo y entenderá la importancia del código limpio para la selección de nombres de identificadores y comentarios en la implementación de los algoritmos.

II. Metodología

El curso es de carácter expositivo con resolución de trabajos prácticos presentados en la computadora. Para el desarrollo del curso se utilizarán recursos audiovisuales y guías de estudio como complemento a lo expuesto durante las clases. En determinadas clases, se hará uso de un equipo de cómputo que permita mostrar diversas situaciones prácticas.

En las sesiones de laboratorio, que podrían tener alguna actividad asincrónica programada, los alumnos resuelven de forma individual, según se indique, un tema preparado por los profesores del curso, el cual se basa en los materiales de enseñanza disponibles para el curso. En estas sesiones, se relacionan los conocimientos teóricos con la práctica y se busca que el estudiante utilice un entorno de desarrollo integrado que permita la implementación de algoritmos usando el paradigma de programación imperativo. Las consultas que el estudiante necesite realizar al profesor del curso las puede hacer durante la clase, en los horarios de asesoría que el profesor proporcione, por correo electrónico o algún otro medio electrónico que indique el profesor del curso.

III. Sumilla

El curso es de naturaleza teórico-práctica y tiene como propósito desarrollar las capacidades para resolver diversos problemas de las ciencias e ingeniería planteando alternativas de solución que utilicen como herramienta la programación imperativa. Se desarrollan las nociones básicas de lógica proposicional con énfasis en la negación, la conjunción y la disyunción. Asimismo, se estudian los conceptos necesarios para el diseño de algoritmos enfatizando la representación de estructuras algorítmicas selectivas e iterativas por medio del diagrama de flujo y el pseudocódigo. Finalmente, se implementarán los algoritmos usando un lenguaje de programación imperativo.

IV. Descripción del programa

CAPÍTULO 1. Lógica proposicional (2 horas)

Lógica proposicional y proposición lógica. Operaciones lógicas: negación, conjunción y disyunción. Tabla de verdad de las operaciones lógicas. Representación de proposiciones lógicas en la programación imperativa. Proposiciones lógicas simples y compuestas. Tautología, contradicción y contingencia.

CAPÍTULO 2. Algoritmia (2 horas)

Algoritmia. Características de los algoritmos. Lenguaje de alto nivel, lenguaje de bajo nivel, lenguaje de máquina. Traducción, compilación e interpretación. Resolución de problemas usando el computador. Algoritmos secuenciales. Representación de algoritmos secuenciales en diagrama de flujo. Representación de algoritmos secuenciales en pseudocódigo. Implementación de algoritmos secuenciales en un programa.

CAPÍTULO 3. Estructura algorítmica selectiva simple y doble (2 horas)

Estructuras de control de flujo. Estructura algorítmica selectiva. Representación de la estructura algorítmica selectiva en diagrama de flujo. Representación de la estructura algorítmica selectiva en pseudocódigo. Implementación de la estructura algorítmica selectiva en un programa. Resolución de problemas usando el computador.

CAPÍTULO 4. Estructuras algorítmicas selectivas anidadas (4 horas)

Estructuras selectivas anidadas. Representación de estructuras selectivas anidadas en diagrama de flujo. Representación de estructuras selectivas anidadas en pseudocódigos. Implementación de estructuras selectivas anidadas en un programa. Resolución de problemas usando el computador.

CAPÍTULO 5. Estructura algorítmica iterativa con entrada controlada (2 horas)

Estructura iterativa con entrada controlada. Representación de la estructura iterativa con entrada controlada en diagrama de flujo. Representación de la estructura iterativa con entrada controlada en pseudocódigos. Implementación de la estructura iterativa con entrada controlada en un programa. Iteración controlada por contador. Resolución de problemas usando el computador.

CAPÍTULO 6. Programación modular (4 horas)

Programación modular. Tipos de subprogramas. Implementación de subprogramas en un lenguaje de programación. Resolución de problemas usando el computador.

CAPÍTULO 7. Control de iteraciones (2 horas)

Iteración controlada por contador. Iteración controlada por centinela. Resolución de problemas usando el computador

CAPÍTULO 8. Estructuras algorítmicas iterativas con salida controlada (2 horas)

Estructura iterativa con salida controlada. Representación de la estructura iterativa con salida controlada en diagrama de flujo. Representación de la estructura iterativa con salida controlada en pseudocódigos. Implementación de la estructura iterativa con salida controlada en un programa. Iteración controlada por contador. Iteración controlada por centinela. Resolución de problemas usando el computador.

CAPÍTULO 9. Estructuras algorítmicas iterativas anidadas (2 horas)

Estructuras iterativas anidadas. Resolución de problemas usando el computador.

CAPÍTULO 10. Diseño estructurado (4 horas)

Diseño estructurado. Etapas del diseño estructurado. Técnica de refinamiento sucesivo. Implementación de diseño estructurado en un lenguaje de programación. Resolución de problemas usando el computador.

CAPÍTULO 11. Depuración (2 horas)

Tablas de datos. Ejecución paso a paso. Resolución de problemas usando el computador.

V. Descripción del laboratorio

- Entorno de desarrollo
- Operadores y lógica
- Estructura básica de un programa
- Programas secuenciales
- Estructura algorítmica selectiva
- Ciclo iterativo con entrada controlada
- Programación modular
- Estructura algorítmica selectiva múltiple y salida controlada
- Ciclos iterativos anidados
- Diseño estructurado y tabla de datos

VI. Bibliografía

- Textos guía

DEITEL, Harvey M.

2004 C: *how to program*. Cuarta edición. Upper Saddle River, N.J.: Prentice-Hall.

Enlace permanente al catálogo de biblioteca:

[https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:\\$002f\\$002fSD_ILS\\$002f0\\$002fSD_ILS:366438/one](https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:$002f$002fSD_ILS$002f0$002fSD_ILS:366438/one)

KERNIGHAN, Brian W.

1991 *El lenguaje de programación C*. Segunda edición. México D.F.: Prentice-Hall.

Enlace permanente al catálogo de biblioteca:

[https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:\\$002f\\$002fSD_ILS\\$002f0\\$002fSD_ILS:190522/one](https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:$002f$002fSD_ILS$002f0$002fSD_ILS:190522/one)

LÓPEZ ROMAN, Leobardo

2005 *Programación estructurada en lenguaje C*. México, D.F.: Alfaomega.

Enlace permanente al catálogo de biblioteca:

[https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:\\$002f\\$002fSD_ILS\\$002f0\\$002fSD_ILS:423904/one](https://pucp.ent.sirsi.net/client/es_ES/campus/search/detailnonmodal/ent:$002f$002fSD_ILS$002f0$002fSD_ILS:423904/one)

- Texto complementario

CORREA URIBE, Guillermo

1992 *Desarrollo de algoritmos y sus aplicaciones en Basic, Pascal, Cobol y C con su respectivo pseudocódigo*. Tercera edición. Bogotá: McGraw-Hill.

VII. Sistema de evaluación

Los promedios de prácticas se calculan con aproximación hasta las décimas. Cualquiera sea la cifra de las centésimas, no se tomará en cuenta.

La nota final del curso se expresa solo en números enteros. Si el cálculo de la nota final da un total con decimales, debe convertirse esa cifra a enteros (se añade un punto a la nota si el primer decimal es cinco o más; se elimina el decimal si es menor de 5).

La nota final del curso se calculará utilizando la fórmula que a continuación se detalla. En ella se usa la siguiente nomenclatura:

N_f : nota final

E_1 : nota del primer examen (medio ciclo)

E_2 : nota del segundo examen (final)

Pb : promedio de prácticas de tipo Pb, eliminando la nota más baja.

$$N_f = \frac{2E_1 + 3E_2 + 5Pb}{10}$$

De acuerdo a las Normas del Ciclo de Verano y las disposiciones aprobadas por Consejo de Estudios Generales Ciencias para el dictado de cursos en los ciclos de verano, no existirá examen especial.

Si alguna clase presencial no pudiera realizarse en la fecha programada u otra posible, por alguna situación extraordinaria, la Secretaría Académica podrá autorizar que se sustituya por una sesión remota o que no se lleve a cabo, lo cual comunicará al estudiante de forma oportuna por correo electrónico.

San Miguel, enero de 2024