

PROGRAMA DE ESTUDIO

A. Antecedentes Generales

ASIGNATURA	: Tecnologías de Información II
CÓDIGO	: III222A
DURACIÓN	: UN SEMESTRE ACADÉMICO
PRE- REQUISITO	: TECNOLOGIAS DE INFORMACION I
CO – REQUISITO	: NO TIENE
UBICACIÓN	: SEGUNDO AÑO, PRIMER SEMESTRE
CARÁCTER	: OBLIGATORIO
HRS. DIRECTAS ASIGNATURA	: 68 - 34
HRS. DIRECTAS SEMANALES	: 4 - 2
CRÉDITOS	: 10

B. Intenciones del Curso

El curso obligatorio de **Tecnologías de Información II**, perteneciente al ciclo de Bachillerato y correspondiente al segundo curso de la línea de Tecnologías de la Información. Este curso permitirá al alumno conocer, comprender y aplicar conceptos relacionados con el desarrollo de aplicaciones computacionales en base a objetos.

Durante el desarrollo del curso se estudia un lenguaje programación de alto nivel, de modo de permitir la creación de aplicaciones.

C. Objetivos Generales

OBJETIVOS FORMATIVOS

En el plano conceptual

- Comprender la estructura de un modelo abstracto a través de algoritmos y diagramas de flujo.

** Este programa puede ser objeto de modificación al inicio del periodo académico**

** This syllabus may be subject to change at the beginning of the semester**

- Identificar la interrelación existente entre los diferentes componentes de un programa computacional.

En el plano procedimental

- Diseñar soluciones informáticas frente a problemas del mundo real.
- Elaborar algoritmos describiéndolos a través de diagramas de flujo tomando como referencia un algoritmo conocido.

En el plano actitudinal

- Reconocer la importancia de los lenguajes de programación como facilitadores del tratamiento de datos para generar información.

C.1. NIVEL CONCEPTUAL

- Comprender los conceptos más importantes que definen un programa computacional.
- Establecer la interrelación que existe entre los diferentes componentes de un programa computacional.
- Comprender el concepto de objeto dentro del contexto de programación computacional, y su aplicación en la solución de problemas.

C.2. NIVEL PROCEDIMENTAL

- Interpretar la realidad de un proceso de la vida real, de diversa índole, y traducirlo en modelos abstractos que representen fielmente el comportamiento de este proceso.
- Aplicar los conceptos más importantes que definen un programa computacional.
- Construir algoritmos y describirlos a través de diagramas de flujo, basándose en un algoritmo conocido.
- Aplicar el algoritmo y el diagrama de flujo en un programa escrito en un lenguaje de programación de alto nivel que respondan efectivamente a los requerimientos planteados.

C.3. NIVEL ACTITUDINAL

- Reconocer la importancia de los lenguajes de programación como facilitadores del tratamiento de datos para generar información.
- Reconocer la necesidad del orden, claridad y rigor en el tratamiento, ordenación y presentación de la información.
- Desarrollar autonomía en la resolución de problemas.
- Desarrollar la capacidad de trabajo en equipo asumiendo los diversos roles asignados.

** Este programa puede ser objeto de modificación al inicio del periodo académico**

** This syllabus may be subject to change at the beginning of the semester**

D. Contenidos

D.1 UNIDAD 1: Diagrama de flujo de datos (DFD)

CONTENIDOS CONCEPTUALES

- Introducción de algoritmos.
- Introducción a diagramas de flujos.
- Concepto de datos y su estructura.
- Control de flujo (sentencias condicionales IF)
- Control de flujo (sentencias condicionales WHILE y FOR)

D.2 UNIDAD 2: Lenguaje de programación orientada a objetos

CONTENIDOS CONCEPTUALES

- Introducción al lenguaje de programación.
- Entrada y salida de datos.
- Tipos y declaración de variables.
- Operadores.
- Sentencias condicionales (IF, SWITCH).
- Ciclos (WHILE, DO WHILE, FOR).
- Arreglos y cadenas.
- Arreglos multidimensionales (matrices).
- Funciones. Pasaje de parámetros.
- Objetos.

E. Metodología

El curso se desarrollará mediante las siguientes estrategias metodológicas:

- Clases expositivas, en que el profesor expondrá a los alumnos la base de los conceptos requeridos para alcanzar los objetivos. Será deber del alumno profundizar e interiorizar dichos conceptos, ya sea mediante la lectura bibliográfica, investigación, práctica y/u otro medio que el alumno considere importante.
- Talleres de resolución de problemas con uso de software, donde los alumnos analizan un caso del mundo real y construyen una solución en lenguaje de programación.

** Este programa puede ser objeto de modificación al inicio del periodo académico**

** This syllabus may be subject to change at the beginning of the semester**

- Desarrollo de un proyecto integral donde los alumnos aplican los conocimientos de programación en un entorno de trabajo colaborativo.

F. Evaluación

F1. EVALUACIÓN CONCEPTUAL Y PROCEDIMENTAL

Para las diferentes instancias evaluativas se contará con una pauta de corrección con criterios claros y conocidos por los alumnos. La pauta será acorde con las exigencias planteadas por el profesor. Lo anterior es válido para los test, certámenes, talleres y trabajos.

1. **Test:** Se realizarán test quincenales.

2. **Talleres de resolución de problemas:** Problemas cortos de solución acotada que serán desarrollados de manera grupal o individual como complemento a los visto por el profesor y en las ayudantías. El desarrollo y término del problema se realizará durante el módulo de clases. Lo anterior significa que la asistencia a los módulos de taller es **obligatoria**. Algunos de los talleres se realizarán con apoyo de software.

3. **Proyecto:** donde los alumnos representan una situación o problema de la vida real mediante la elaboración de un programa en lenguaje C++. Como producto del proyecto los alumnos deben entregar un ejecutable y un informe final que incluya una descripción de la situación a analizar, el diagrama de flujo y el algoritmo de programación.

4. **Certámenes:** Se realizarán dos certámenes, en las semanas establecidas por la Facultad.

5. **Examen:** Se llevará a cabo al término del semestre, en la fecha establecida por la Facultad, y exigiéndose nota mínima de 3.0, para todos los alumnos, según el R.A.A.R.

La ponderación de las diferentes instancias de control (de contenido, procedimental, actitudinal) en la nota final del alumno se desglosa de la siguiente manera:

- 20 % Certamen 1.
- 25 % Certamen 2.
- 15 % Controles-tareas (o equivalentes)
- 10 % Proyecto.

** Este programa puede ser objeto de modificación al inicio del periodo académico**

** This syllabus may be subject to change at the beginning of the semester**

- 30 % Examen

F2. EVALUACIÓN ACTITUDINAL

Los alumnos deben ser capaces no sólo de resolver los problemas, sino que deben ser capaces de comunicar la solución en un lenguaje técnico adecuado, de acuerdo a los conceptos y nomenclatura entregados en clases.

G. Bibliografía

OBLIGATORIA

- KERNIGHAN, B., “**EL LENGUAJE DE PROGRAMACION C**”, PRENTICE – HALL HISPANOAMERICANA, 1991.
- JOYANES, L., “**FUNDAMENTOS DE PROGRAMACIÓN: ALGORITMOS Y ESTRUCTURA DE DATOS**”, 4ª ED. MCGRAW HILL, 2008.

COMPLEMENTARIA

- STROUSTRUP, B., “**EL LENGUAJE DE PROGRAMACION C++**”, 3ª ED, EDDISON-WESLEY IBEROAMERICANA, 1997.

** Este programa puede ser objeto de modificación al inicio del periodo académico**

** This syllabus may be subject to change at the beginning of the semester**