

Plan Global — Elementos de Programación y Estructuras de Datos — Facultad de Ciencias y Tecnología — UMSS

Febrero, 2024

1. Datos de Identificación

Nombre de la Materia	Elementos de Programación y Estructuras de Datos
Código	2010003
Semestre	2do semestre
Carga horaria	4 periodos Teóricos, 2 periodos practica
Materias Relacionadas	Introducción a la Programación (2010010) Métodos Técnicas y Taller de Programación (2010012)
Docente	Lic. Helder Octavio Fernández Guzmán
Correo Electrónico	h.fernandez@umss.edu

2. Justificación

2.1. RAZON DE SER DE LA ASIGNATURA

La materia de Elementos de programación y Estructuras de datos[1] propone metodologías de programación elaboradas para descomponer y resolver problemas de forma elegante, eficiente y correcta.

Adicionalmente contempla el estudio de estructuras de datos, para conocer como representar, organizar y gestionar información compleja.

El estudio de esta materia permite conocer más a fondo los conceptos que identifican elementos importantes que enriquecen la metodología de programación, también permite tratar mecanismos para representar, ordenar y organizar datos en la computadora de tal manera que puedan utilizarse de manera más óptima.

2.2. DENTRO DEL PLAN DE ESTUDIOS

A nivel práctico y en el medio, compañías e industrias del area tecnológica contemplan el desarrollo de software y usan herramientas y métodos de programación óptimos, principalmente se concentran en el uso de Lenguajes Orientado a Objetos para la elaboración de

programas debido a las ventajas que se obtienen tanto para la elaboración de soluciones robustas y eficientes, así como para aprovechar las ventajas respecto a la reutilización de código.

Conocer técnicas de programación avanzadas, mismas que se proponen en la materia, puede representar la posibilidad de adquirir competencias significativas desde el punto de vista profesional. La programación es una de las habilidades más importantes que tienen profesionales del área de Ingeniería.

2.3. COMO CONTRIBUCIÓN A LA FORMACIÓN INTEGRAL DEL PROFESIONAL

La materia proporciona un conjunto de técnicas y elementos, que son muy utilizados en la práctica, lo que le permitirá elaborar programas eficientes y de mayor calidad. También estará capacitado para integrarse a equipos multidisciplinarios de trabajo, para poder aportar con conocimientos especializados en Informática muy aplicables en todo campo. Además, el estudiante tendrá conocimientos que lo capacitan para seguir mejorando y conociendo nuevos elementos y herramientas para el diseño y gestión de datos e información.

La materia de Elementos de Programación y Estructuras de datos propuesta en el presente plan propone cimentar en los conceptos de: Programación orientada a objetos, Estructuras de Datos y Técnicas de programación.

Al finalizar el curso se espera que el estudiante presente soluciones correctas a problemas de dimensión mediana o compleja, utilizando las estructuras de datos más convenientes y considerando nuevos elementos de programación que le permitan tener una solución que sea elegante y óptima en términos de eficiencia y calidad de código.

3. Objetivos

Con el desarrollo de los contenidos de la materia se busca en el estudiante:

- Habilidades para enfrentar la solución de problemas utilizando: técnicas algorítmicas de programación basadas en recursión, programación bajo el paradigma Orientado a Objetos y sus características avanzadas.
- Comprensión de los conceptos de administración de memoria durante la ejecución de los programas con el fin de implementar estructuras y soluciones basadas en estos conceptos, de tal forma que se adecúen de la forma más idónea a los recursos utilizados.
- Conocer e identificar las estructuras de datos, de forma conceptual y práctica.
- Aplicación de las estructuras de datos en la resolución de problemas de manera óptima y de acuerdo a las necesidades particulares del caso.

4. Selección y Organización de Contenidos

4.1. UNIDAD 1

UNIDAD	PROGRAMACIÓN ORIENTADA A OBJETOS
Objetivo	Al finalizar la unidad el estudiante tendrá un entendimiento global de la POO, retomando lo aprendido en Introducción a la Programación, fortaleciendo y enriqueciendo los conceptos de herencia y polimorfismo.
Contenido	<ol style="list-style-type: none">1. Conceptos de POO2. Polimorfismo3. Genericidad4. Propiedades de datos abstractos5. Generalización: a través de herencia simple, Clases abstractas e interfaces.6. Librerías y Clases Disponibles en el Lenguaje de Programación (ej. Entrada y salida)7. Elaboración de un programa haciendo énfasis en la POO y la Generalización.

4.2. UNIDAD 2

UNIDAD	ELEMENTOS DE PROGRAMACIÓN
Objetivo	Al finalizar la unidad el estudiante tendrá la capacidad de plantear soluciones recursivas, considerando bases matemáticas de diseño, asegurando la correctitud de sus soluciones.
Contenido	<ol style="list-style-type: none">1. Definición de Recursividad [2].2. Reglas de la Recursividad. Mecanismos formales de recursión.3. Uso de la Recursividad como alternativa a las Iteraciones.4. Costos del uso de la Recursividad.5. Elaboración de ejemplos de aplicación en código.6. Ordenamiento y búsqueda.7. Permutaciones, combinaciones.8. Back tracking.9. Implementación de la recursividad en ejemplos de aplicación.

4.3. UNIDAD 3

UNIDAD	ADMINISTRACIÓN y ASIGNACIÓN DE LA MEMORIA EN JAVA
Objetivo	Al final de la Unidad el estudiante será capaz de diferenciar entre tipos de datos estáticos y dinámicos, sus ventajas y desventajas, así como comprender los mecanismos de administración de memoria para poder proveer programas orientados a la alta performance y con manejo de la degradación.
Contenido	<ol style="list-style-type: none">1. Áreas de la memoria.2. Gestión de memoria dinámica.3. Miembros estáticos.4. variables y referencias.5. la palabra reservada This.6. Operaciones con referencias o enlaces7. Clonación

4.4. UNIDAD 4

UNIDAD	PROGRAMACIÓN GENÉRICA
Objetivo	Al finalizar la unidad los estudiantes serán capaces de utilizar el concepto de programación genérica y la forma de implementarlo. Hacer buen uso del principio de abstracción para proveer de implementaciones independientes del tipo de datos.
Contenido	<ol style="list-style-type: none">1. El tipo Object.2. Concepto de programación genérica.3. Clases genéricas - templates. Genéricas en Estructuras de Datos.

4.5. UNIDAD 5

UNIDAD	ESTRUCTURAS DE DATOS LINEALES
Objetivo	Al final de la Unidad, el estudiante será capaz de explicar, especificar, distinguir y aplicar las estructuras de datos[3] lineales en la solución de problemas programáticos.
Contenido	<ol style="list-style-type: none">1. Definición de estructuras de datos y TDA.2. Estructuras de Datos Lineales.3. Operaciones estandar con listas enlazadas.4. Tipos abstractos de datos: listas, pila, cola5. Recursos disponibles en el Lenguaje de Programación.6. Elaboración de Programas usando estructuras de datos lineales.

4.6. UNIDAD 6

UNIDAD	ESTRUCTURAS DE DATOS NO LINEALES
Objetivo	Al final de la unidad, el estudiante tendrá la habilidad para especificar y programar estructuras de datos no lineales, árboles, además de aplicar su uso en la resolución de problemas.
Contenido	<ol style="list-style-type: none">1. Definición de estructuras de datos no lineales.2. Árboles.3. Árboles n- arios.4. Árboles binarios (de búsqueda, binarios, B, balanceados)5. Operaciones con árboles: inserción, borrado, recorrido, búsqueda.6. Análisis de desempeño.7. Grafos: dirigidos – no dirigidos; recorridos, búsquedas, conectividad, caminos.8. Elaboración de programas usando estructuras de datos no lineales.

5. Metodología

Clases interactivas donde se van ilustrando los conceptos con ejemplos. Se reforzarán los contenidos observados a través de la propuesta de desarrollo de soluciones por parte del Estudiante (aplicando los diferentes conceptos propuestos).

El éxito de la materia radica en la práctica en computadora dentro y fuera de clases por

parte del estudiante por lo que, por cada unidad, el estudiante deberá realizar trabajos de aplicación utilizando de forma correcta los conceptos. La entrega de estos trabajos es obligatoria y es considerada dentro de la evaluación (40 % del puntaje en las evaluaciones parciales).

Adicionalmente y con el objetivo de asegurar una correcta manipulación de las herramientas usadas en la materia además de verificar el funcionamiento de implementaciones elaboradas en clase, las primeras clases se revisa el correcto uso de las TACs que se usan en la materia (p. ej. laboratorios virtuales).

Finalmente se buscará un reforzamiento de los contenidos a través de material adicional de apoyo, mismo que se facilitará a los estudiantes en la plataforma Moodle de manera coordinada.

6. Cronograma

UNIDAD	Horas Academicas
1. Programación Orientada a Objetos	20
2. Elementos de Programación	30
3. Administración y asignación de la Memoria	10
4. Programación Genérica	12
5. Estructuras de Datos Lineales	24
6. Estructuras de Datos No Lineales	24
TOTAL	120

7. Criterios de Evaluación

Proceso de evaluacion permanente

Evaluación diagnóstica. Al principio del semestre y al inicio de cada unidad, para identificar la base de conocimiento del estudiante respecto a los temas a trabajar. En base al resultado de esta evaluación se podría realizar un repaso de los conceptos más importantes para nivelar el conocimiento y con el propósito de obtener mejores resultados en la asimilación de los nuevos contenidos propuestos.

Evaluación formativa. Por cada unidad el estudiante deberá resolver cuestionarios y trabajos de laboratorio donde se aplican los conceptos examinados.

Evaluación sumativa. La materia consta de evaluaciones parciales que tienen un ponderado en la nota final, así mismo los trabajos que se realizan en el transcurso del semestre serán de carácter habilitante a estas pruebas escritas.

Evaluaciones formales

Para todas las evaluaciones consideradas formales, se pretende regirse a las normas de evaluación usadas en la Universidad, buscando siempre involucrar al estudiante en el proceso de evaluación de tal manera que sea posible detectar a tiempo falencias y poder subsanarlas. La evaluación de la materia es a través de desarrollo de prácticas y exámenes de laboratorio o prueba escrita. La nota se dividirá en dos partes: actividades prácticas y exámenes.

1. La evaluación del primer y segundo parcial; tendrá la siguiente ponderación:
 - a) Evaluación formativa: 20 %, prácticas y evaluaciones cortas durante el semestre dentro las cuales pueden estar consideradas actividades de auxiliatura.
 - b) Evaluación sumativa: 80 %
2. La evaluación del examen final, tendrá la siguiente ponderación:
 - a) Evaluación sumativa: 100 %. Para poder dar este examen, debe haber participado activamente de las actividades y laboratorios durante el semestre.
3. La evaluación del examen de segunda instancia es solamente sumativa con una ponderación del 51 %. Solo las personas con un promedio de los parciales mayor o igual a 26 puntos en los 2 primeros parciales pueden habilitarse para la evaluación de 2da Instancia.

Referencias

- [1] Joyanes Aguilar Luis. *Estructuras de Datos en Java*. Mc Graw Hill, 1 edition, 2008.
- [2] S Roberts Erick. *Thinking Recursively*. John Wiley & Sons Inc., 1 edition, 1986.
- [3] Mark Allen Weiss. *Data Structures and Algorithm Analysis in Java*. Pearson Education, 1 edition, 2012.