

PLAN GLOBAL
ELEMENTOS DE PROGRAMACIÓN Y ESTRUCTURAS DE DATOS

I. DATOS DE IDENTIFICACIÓN

■ Nombre de la materia:	Elementos de Programación y Estructuras de Datos
■ Código:	2010003
■ Grupo:	1, 2, 3, 4 y 5
■ Carga horaria:	4 hrs teoría, 2 hrs práctica
■ Materias con las que se relaciona:	Introducción a la Programación Métodos y Técnicas de Programación
■ Docente:	Leticia Blanco Rose Mary Salazar
■ Teléfono:	4233719
■ Correo Electrónico:	leticia@memi.umss.edu.bo rsalazar@umss.edu.bo

II. JUSTIFICACIÓN

Después de tener los conocimientos básicos de programación, se necesita contar con más elementos que ayuden a un profesional del área de ciencias de la computación, a resolver problemas de forma elegante, eficiente y correcta, la finalidad de este curso es brindar al estudiante un conjunto de conceptos que identifican elementos importantes que enriquecen la metodología de programación, es importante también considerar las distintas estructuras que existen para representar los datos.

Por otro lado, la administración de datos es importante, por lo que se estudian estructuras de datos, añadiendo más conocimientos sobre la forma de programar soluciones, los mismos que son una antesala a estructuras de datos más complejas que se utilizan en la resolución de problemas de mayor envergadura.

El presente plan propone introducir los conceptos de: programación orientada al objeto considerando genéricas, técnicas de programación, recursión y estructuras de datos.

Al finalizar se espera que los estudiantes presenten soluciones correctas a pequeños problemas, utilizando las estructuras de datos adecuadas y considerando nuevos elementos de programación que le permitan tener una solución elegante y eficiente.

III. OBJETIVOS

Los objetivos de esta materia son:

- Conseguir habilidades para enfrentar la solución de problemas utilizando la POO
- Introducir a las estructuras de datos, de forma conceptual y práctica
- Aplicar estructuras de datos adecuadas para la resolución de problemas
- Introducir concepto de administración dinámica de memoria e implementar estructuras de datos en base a enlaces

IV. SELECCIÓN Y ORGANIZACIÓN DE CONTENIDOS

UNIDAD 1: ELEMENTOS DE PROGRAMACIÓN

Objetivo de la Unidad

- Al finalizar la unidad, el estudiante será capaz de: plantear soluciones recursivas, considerando bases matemáticas de diseño, asegurando la correctitud de su solución.

Contenido:

- 1.1 Definición de recursión.
- 1.2 Partes de la recursión.
- 1.3 Reglas de recursividad.
- 1.4 Recursión versus iteración
- 1.5 Coste espacial
- 1.6 Implementación de la recursividad en ejemplos de aplicación.

UNIDAD 2: PROGRAMACIÓN ORIENTADA AL OBJETO

Objetivo de la Unidad

- Al final de la unidad, el estudiante tendrá un marco general de la POO, retomando lo aprendido en Introducción a la Programación y fortaleciendo conceptos de herencia y manejo de flujos de JO.

Contenido:

- 2.1 Conceptos básicos de la POO
- 2.2 Polimorfismo
- 2.3 Genericidad
- 2.4 Propiedades de datos abstractos
- 2.5 Generalización: estructural y de facilidades – herencia. Clases abstractas, interfaces
- 2.6 Flujos de Entrada y salida

UNIDAD 3: MEMORIA DINÁMICA

Objetivo de la Unidad

- Al final de la unidad el estudiante será capaz de analizar cuáles son las ventajas e inconvenientes de esta técnica, distinguir las diferencias entre tipos de datos estáticos y dinámicos en cuanto a su almacenamiento en memoria principal y su tratamiento por parte de los programas (algoritmos) que los manejan. Así mismo podrá explicar el funcionamiento de la gestión dinámica de memoria.

Contenido:

- 3.1 Datos estáticos y dinámicos.
- 3.2 Tipo puntero - referencia.
- 3.3 Gestión de memoria dinámica
- 3.4 Operaciones con punteros.
- 3.5 Listas enlazadas.
- 3.6 Operaciones con listas enlazadas.
- 3.7 Listas doblemente enlazadas.

UNIDAD 4: ESTRUCTURAS DE DATOS LINEALES

Objetivo de la Unidad

- Al final de la unidad, el estudiante será capaz de: explicar, especificar, distinguir y aplicar las estructuras de datos lineales.

Contenido:

- 4.1 Definición y representación de estructuras de datos lineales.
- 4.2 Tipo abstracto de dato: pila, cola.
- 4.3 Tipos de estructuras de almacenamiento.
- 4.4 Operaciones con pila.
- 4.5 Operaciones con colas.
- 4.6 Paquete de estructuras de datos lineales.
- 4.7 Ejercicios de aplicación.

UNIDAD 5: ESTRUCTURAS DE DATOS NO LINEALES

Objetivo de la Unidad

- Al final de la unidad, el estudiante será capaz de especificar y programar estructuras de datos no lineales, árboles, además de aplicar su uso en la resolución de problemas.

Contenido:

- 5.1 Definición de estructuras de datos no lineales.
- 5.2 Árboles.
- 5.3 Árboles n-arios. 5.4 Árboles binarios.
- 5.5 Árboles binarios de búsqueda.
- 5.6 Árboles balanceados. 5.7 Árboles B.
- 5.8 Operaciones con árboles: inserción, borrado, recorrido, búsqueda.
- 5.9 Análisis de desempeño.
- 5.10 Grafos: dirigidos – no dirigidos; recorridos, búsquedas, conectividad, caminos.
- 5.11 Ejercicios de aplicación.

V. METODOLOGIAS

- Clases magistrales ilustrando los conceptos con ejemplos. El estudiante tiene que implementar aplicaciones pequeñas utilizando los diferentes conceptos de la materia.
- Es importante que el estudiante practique en la computadora fuera de clases, por cada unidad, el estudiante debe entregar trabajos de aplicación utilizando de forma correcta los conceptos. Estos trabajos son imprescindibles para rendir los exámenes escritos.
- Por otro lado, para verificar el funcionamiento de implementaciones de clase, se plantean algunas de las clases en laboratorio.
- Material de apoyo se provee a través de la página de la materia www.memi.umss.edu.bo/elementos, las herramientas para este curso son Java y BlueJ.

VI. CRONOGRAMA O DURACIÓN EN PERIODOS ACADÉMICOS POR UNIDAD

UNIDAD	DURACIÓN (HORAS ACADÉMICAS)	DURACIÓN EN SEMANA
Elementos de programación	18	3
Programación orientada al objeto	18	3
Memoria Dinámica	18	3
Estructuras de datos lineales	24	4
Estructuras de datos no lineales	30	5

VII. CRITERIOS DE EVALUACIÓN

- Evaluación diagnóstica. Al principio del semestre es necesario nivelar a los estudiantes respecto a conceptos de POO. Para ello se hace un repaso de los conceptos principales, en base a la participación de los estudiantes respondiendo a preguntas y respuestas que el profesor realiza. Del mismo modo en clases acerca de material ya avanzado.
- Evaluación formativa. Durante el semestre y por cada unidad el estudiante debe entregar trabajos que apliquen los conceptos introducidos, esta tarea se hace en coordinación con los auxiliares de la materia.
- Evaluación sumativa. La materia consta de exámenes escritos que tienen un ponderado en la nota final, así mismo los trabajos que se realizan en el transcurso del semestre son de carácter habilitante a estas pruebas escritas.
- En términos generales, la evaluación de la materia es a través de desarrollo de prácticas y exámenes escritos.
- La nota se divide en dos porciones: prácticas y exámenes.
- Las prácticas se desarrollan con el auxiliar y el docente, las prácticas tienen un valor de 10 puntos sobre los 100. Los docentes y auxiliares darán prácticas la mismas que son evaluadas sobre 10 puntos. Las prácticas serán de acuerdo al material que se avance.
- Los exámenes escritos serán sobre 90 puntos. Esta modalidad es válida para los exámenes de primer, segundo parcial y examen final, quedando fuera de esta regla la segunda instancia la cual por si misma tiene normas especiales y de excepción, como es tener un promedio de los parciales mayor a 25 puntos.

VIII. BIBLIOGRAFÍA

Texto base:

- Thinking Recursively, Eric S. Roberts, John Wiley & Sons, Inc. 1986
- Estructuras de datos y algoritmos, MARK ALLEN WEISS. Addison Wesley 2nd edition, 2006