

q

1.- DATOS DE LA ASIGNATURA

| | |
|---------------------------|---|
| Nombre de la asignatura : | Estructura de Datos |
| Carrera : | Ingeniería en Animación Digital y Efectos Visuales |
| Clave de la asignatura : | AVD-1210 |
| SATCA ¹ | 2-3-5 |

2.- PRESENTACIÓN

Caracterización de la asignatura.

Esta asignatura aporta al perfil del Ingeniero en Animación Digital y Efectos Visuales:

- Aplica los lenguajes de programación utilizados en el desarrollo de software para la industria de la animación.
- Desarrolla herramientas de software para facilitar y optimizar los procesos de producción de la animación digital y los efectos visuales.
- Desarrolla modelos de simulación física en entornos gráficos computacionales aplicables a proyectos de animación y efectos visuales.

Puesto que esta asignatura es base fundamental del perfil del egresado y relacionadas con todas aquellas relacionadas con programación.

Intención didáctica.

Se organizan los contenidos en cinco temas, iniciando con el tema uno donde se conceptualizan las estructuras de datos así como su clasificación.

El segundo tema, aborda el conocimiento acerca del uso de algoritmos para creación y manejo de listas enlazadas.

El manejo de arboles, recursividad y grafos es el tema de estudio tres.

Algoritmos de ordenamiento, búsqueda y recuperación de datos son el tema cuatro.

Para concluir el curso en el tema cinco se realiza un estudio sobre el análisis de la complejidad y eficiencia de los algoritmos, lo cual permitirá determinar cuáles son los algoritmos más eficientes para solucionar un problema.

Se sugiere una actividad integradora con el objeto de reforzar y de evidenciar lo aprendido en el curso; en esta actividad integradora se propone la programación de un sistema que de una solución a una problemática real, buscando que el estudiante tenga contacto con los conceptos en forma concreta y sea a través del análisis, creatividad e imaginación que se logre tal objetivo. Esta actividad integradora se propone que se defina en el transcurso del tema uno de esta competencia.

¹ Sistema de Asignación y Transferencia de Créditos Académicos

En el transcurso de las actividades programadas, es muy importante que el estudiante aprenda a valorar las actividades que lleva a cabo y entienda que está construyendo su futuro y en consecuencia actúe de una manera profesional; de igual manera, aprecie la importancia del conocimiento y los hábitos de trabajo; desarrolle la capacidad de análisis, precisión y la curiosidad, la puntualidad, el entusiasmo y el interés, la tenacidad, la flexibilidad y la autonomía.

3.- COMPETENCIAS A DESARROLLAR

Competencias específicas:

- Aplica eficientemente las estructuras de datos en la elaboración de programas relacionados a la animación digital y efectos visuales, teniendo en cuenta los métodos de ordenamiento y búsqueda para la optimización del rendimiento de dichos programas.

Competencias genéricas:

Competencias instrumentales

- Capacidad de análisis y síntesis.
- Capacidad de organizar y planificar.
- Habilidades básicas de manejo de la computadora.
- Habilidad para buscar y analizar información proveniente de fuentes diversas.
- Solución de problemas.
- Toma de decisiones.

Competencias interpersonales

- Capacidad crítica y autocrítica.
- Trabajo en equipo.
- Habilidades interpersonales.

Competencias sistémicas

- Capacidad de aplicar los conocimientos en la práctica.
- Habilidades de investigación.
- Capacidad de aprender.
- Capacidad de generar nuevas ideas (creatividad).
- Habilidad para trabajar en forma autónoma.
- Búsqueda del logro.

4.- HISTORIA DEL PROGRAMA

| Lugar y fecha de elaboración o revisión | Participantes | Evento |
|---|---|--|
| Instituto Tecnológico Superior de Chapala, del 23 al 26 de abril de 2012. | Representantes de los Institutos Tecnológicos de: Aguascalientes, Bahía de Banderas, Colima, Cd. Guzmán, La Laguna, La Paz, León, Pachuca, Puebla, San Luis Potosí, Villahermosa, Zacatepec, Superior de Chapala, Estudios Superiores de Ecatepec, Superior de Zapotlanejo y KAXAN Media Group. | Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de la Carrera de Ingeniería en Animación Digital y Efectos Visuales. |
| Desarrollo de Programas de Competencias Profesionales por los Institutos Tecnológicos del 27 de abril al 6 de agosto de 2012. | Academias de la Ingeniería en Animación Digital y Efectos Visuales de los Institutos Tecnológicos de: Aguascalientes, Colima y Villahermosa. | Elaboración del programa de estudio propuesto en la Reunión Nacional de Diseño Curricular de la Carrera de Ingeniería en Animación Digital y Efectos Visuales. |
| Instituto Tecnológico Superior de Chapala, del 7 al 10 de agosto de 2012. | Representantes de los Institutos Tecnológicos de: Aguascalientes, Bahía de Banderas, Colima, Cd. Guzmán, La Laguna, La Paz, León, Pachuca, Puebla, San Luis Potosí, Villahermosa, Zacatepec, Superior de Chapala, Estudios Superiores de Ecatepec, Superior de Zapotlanejo y KAXAN Media Group. | Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de la Carrera de Ingeniería en Animación Digital y Efectos Visuales. |

5.- COMPETENCIAS PREVIAS

- Analiza, diseña y desarrolla soluciones de problemas reales utilizando algoritmos computacionales para implementarlos en un lenguaje de programación.
- Diseña e implementa objetos de programación que permitan resolver situaciones reales y de ingeniería.

6.- TEMARIO

| | Temas | Subtemas |
|----|------------------------------------|--|
| 1. | Fundamentos de estructura de datos | 1.1. Definición 1.2. Clasificación 1.3. Estructuras lineales y no lineales 1.4. Estructuras dinámicas y estáticas |
| 2. | Estructuras lineales | 2.1. Listas 2.2. Pilas estáticas y dinámicas 2.3. Colas estáticas y dinámicas 2.4. Aplicaciones |
| 3. | Estructuras no lineales | 3.1. Recursividad 3.2. Árboles 3.3. Grafos |
| 4. | Métodos de ordenamiento y búsqueda | 4.1. Algoritmos de ordenamiento 4.2. Métodos de búsqueda 4.3. Recuperación de datos |
| 5. | Análisis de los algoritmos | 5.1 Complejidad en el tiempo 5.2 Complejidad en el espacio 5.3 Eficiencia de los algoritmos |

7.- SUGERENCIAS DIDÁCTICAS

El docente debe:

- Ser conocedor de la disciplina que está bajo su responsabilidad, conocer su origen y desarrollo histórico para considerar este conocimiento al abordar los temas. Desarrollar la capacidad para coordinar y trabajar en equipo; orientar el trabajo del estudiante y potenciar en él la autonomía, el trabajo cooperativo y la toma de decisiones. Mostrar flexibilidad en el seguimiento del proceso formativo y propiciar la interacción entre los estudiantes. Tomar en cuenta el conocimiento de los estudiantes como punto de partida y como obstáculo para la construcción de nuevos conocimientos.
- Propiciar actividades de metacognición. Ante la ejecución de una actividad, señalar o identificar el tipo de proceso intelectual que se realizó: una identificación de patrones, un análisis, una síntesis, la creación de un heurístico, etc. Al principio lo hará el docente, luego será el estudiante quien lo identifique.
- Coadyuvar actividades de búsqueda, selección y análisis de información en distintas fuentes.
- Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración entre los estudiantes. Ejemplo: Realizar prácticas en equipo que permitan obtener un resultado a partir del trabajo de todos.
- Relacionar los contenidos de esta asignatura con las demás del plan de estudios, enfatizando a las que ésta da soporte para desarrollar una visión interdisciplinaria en el estudiante.
- Fomentar actividades relacionadas con la lectura, la escritura y la expresión oral que permitan el desarrollo de capacidades intelectuales. Ejemplos: trabajar las actividades prácticas a través de guías escritas, redactar reportes e informes de las actividades de experimentación, exponer al grupo las conclusiones obtenidas durante las observaciones.
- Hacer planteamientos de problemas de su entorno que permita plantear soluciones mediante el uso de estructuras de datos y programe la solución utilizando el lenguaje de programación.
- Encausar el desarrollo de actividades intelectuales de inducción, -deducción, análisis, y síntesis, que encaminen hacia una posición crítica del estudiante.
- Desarrollar actividades de aprendizaje que propicien la aplicación de los conceptos, modelos y metodologías que se van aprendiendo en el desarrollo de la asignatura.
- Plantear problemas que permitan al estudiante la integración de contenidos de la asignatura y entre distintas asignaturas, para su análisis y solución.
- Fomentar el uso de las Nuevas tecnologías de la información y comunicación (NTIC).

8.- SUGERENCIAS DE EVALUACIÓN

La evaluación debe ser continua y formativa por lo que se debe considerar el desempeño en cada una de las actividades de aprendizaje, haciendo especial énfasis en:

- Reportes escritos de las observaciones hechas durante las actividades realizadas en el laboratorio, así como de las conclusiones que de ellas emanen.
- Información obtenida durante las investigaciones solicitadas plasmada en documentos escritos y/o archivos digitales que avalen dicha investigación.
- Exámenes escritos y/o prácticos para comprobar el manejo de aspectos teóricos y declarativos en cada tema.
- Desarrollo de un proyecto final que integre los elementos descritos en el contenido del curso todos los temas de aprendizaje y que de solución a una problemática real.
- Realizar un cuadro comparativo que represente los diferentes algoritmos para la solución de un problema dado, justificando su elección.
- Descripción de otras experiencias concretas que se obtienen al participar en discusiones, exposiciones o cualquier otro medio didáctico-profesional que trate sobre la materia y que debe realizarse durante el curso académico.

9.- TEMAS DE APRENDIZAJE

Tema 1: Fundamentos de estructuras de datos

| <i>Competencia específica a desarrollar</i> | <i>Actividades de Aprendizaje</i> |
|---|--|
| Identifica los conceptos básicos de las estructuras de datos para aplicarlas en el desarrollo de rutinas computacionales. | <ul style="list-style-type: none">• Investigar los conceptos fundamentales de las estructuras de datos.• Identificar las estructuras de datos lineales y no lineales de acuerdo al problema a resolver.• Identificar las estructuras de datos estáticas y dinámicas de acuerdo al problema a resolver. |

Tema 2: Estructuras lineales

| <i>Competencia específica a desarrollar</i> | <i>Actividades de Aprendizaje</i> |
|---|---|
| Aplica las principales estructuras | <ul style="list-style-type: none">▪ Elaborar mapas conceptuales y/o |

| | |
|--|---|
| de datos lineales en la solución de problemas afines al área de la animación digital y efectos visuales. | <p>presentaciones digitales (multimedia) que permitan comprender los conceptos básicos, el funcionamiento y las aplicaciones que tienen las estructuras de datos lineales.</p> <ul style="list-style-type: none"> Realizar ejercicios implementando estructuras de datos lineales. |
|--|---|

Tema 3: Estructuras no lineales

| <i>Competencia específica a desarrollar</i> | <i>Actividades de Aprendizaje</i> |
|---|---|
| Aplica las principales estructuras de datos no lineales para la clasificación y búsqueda de la información. | <ul style="list-style-type: none"> Elaborar mapas conceptuales para comprender los conceptos básicos, el funcionamiento y las aplicaciones que tienen las estructuras de datos no lineales. Realizar ejercicios que implementen soluciones recursivas a soluciones iterativas y viceversa. Realizar ejercicios implementando estructuras de datos no lineales. |

Tema 4: Métodos de ordenamiento y búsqueda

| <i>Competencia específica a desarrollar</i> | <i>Actividades de Aprendizaje</i> |
|--|---|
| Clasifica técnicas para recuperación de información en dispositivos de almacenamiento primario y secundario para gestionar datos de forma óptima y facilitar su procesamiento. | <ul style="list-style-type: none"> Investigar los diversos algoritmos de los métodos de ordenamiento, búsqueda y recuperación de datos según el tipo de problema que se desea resolver. Discutir el uso de los métodos de ordenamiento, búsqueda y recuperación de datos en memoria principal y secundaria. Elaborar un mapa conceptual que represente las diferencias entre los métodos. Aplicar los algoritmos investigados en dos lenguajes orientados a objeto y anotar observaciones. Elaborar una aplicación informática (software) donde se implementen el uso archivos y aplicar métodos de ordenamiento, búsqueda y |

| | |
|--|--|
| | recuperación de datos en memoria principal y secundaria. |
|--|--|

Tema 5: Análisis de los algoritmos

| <i>Competencia específica a desarrollar</i> | <i>Actividades de Aprendizaje</i> |
|---|--|
| Comprende la complejidad de los algoritmos para identificar la eficiencia de los mismos en diferentes situaciones de uso. | <ul style="list-style-type: none"> • Generar cuadro comparativo con los tiempos obtenidos en cada método de ordenamiento y búsqueda. • Emitir su propia conclusión al observar la complejidad de los algoritmos estudiados y contrastarla con lo que dicen al respecto, los autores citados en las fuentes bibliográficas. |

10.- FUENTES DE INFORMACIÓN

1. Joyanes Aguilar, Luis. (2007). *Estructura de Datos en Java*. Madrid, España. Ed. McGraw Hill. Primera edición.
2. Lewis, John. (2007). *Estructura de Datos con JAVA: Diseño de estructuras y algoritmos*. Madrid, España. Ed. Pearson. Primera edición.
3. Guardati Buemo, Silvia. (2007). *Estructura de Datos orientada a objetos: Algoritmos con C++*. México. Ed. Pearson. Primera edición.
4. Allen, Marc. (2007). *Estructura de Datos con JAVA: Compatible con JAVA 2*. Ed. Prentice Hall.
5. Cairo, Osvaldo. (2006). *Estructura de Datos*. Tercera edición. Ed. McGraw Hill.
6. Weiss, M.A. (2011). *Data Structures and Algorithm Analysis in Java, 3rd Edition*, Pearson/Addison Wesley.
7. Hernández, Z.J. y otros. (2005). *Fundamentos de Estructuras de Datos. Soluciones en Ada, Java y C++*, Thomson.
8. Weiss, M.A. (2000). *Estructuras de datos en Java*, Pearson/Addison Wesley.
9. Deitel, P.J. y Deitel, H.M. (2008). *Java. Cómo programar (7ª edición)*, Prentice Hall.
10. Campos Laclaustra, J. (1995). *Estructuras de Datos y Algoritmos*, Prensas Universitarias de Zaragoza, Colección Textos Docentes.
11. Franch Gutiérrez, X. (2001). *Estructuras de Datos. Especificación, Diseño e Implementación*, 3ª edición, Ed. Edicions UPC.
12. Mehta, D.P. y Sahni, S. (2005). *Handbook of Data Structures and Applications*, Chapman & Hall/CRC.

11.- PRÁCTICAS PROPUESTAS

Es recomendable la realización de prácticas en todos los temas que consistan en el modelado y resolución de problemas utilizando lenguajes de programación de actualidad y orientado a objetos; la entrega al final de cada tema de un proyecto que refuerce en la aplicación de los temas vistos en clase y la entrega de un proyecto final que converja en la aplicación de los conceptos vistos en la asignatura, el cual debe ser definido al final del primer tema.

Algunos problemas propuestos:

- Resolver problemas implementando las diversas estructuras de datos lineales en forma estática y en forma dinámica.
- Implementar una función que recibe una lista de enteros L y un número entero n de forma que modifique la lista mediante el borrado de todos los elementos de la lista que tengan este valor.
- Implementar una función *Mezcla2* que tenga como parámetros dos listas de enteros ordenados de menor a mayor y que devuelva una nueva lista como unión de ambas con sus elementos ordenados de la misma forma.
- Resolver lo siguiente: se tienen dos pilas que contienen 12 números enteros; la primera ordenada ascendentemente del 1 al 12 desde el tope hacia el fondo y la segunda ordenada descendentemente del 24 al 13 desde el tope hacia el

fondo, elabore un algoritmo que fusione ambas pilas en una tercera ordenada descendientemente desde el tope hacia el fondo.

- Simular la lógica de una pila utilizando dos colas.
- Simular la lógica de una cola usando dos pilas.
- Escriba un algoritmo de un programa que lea por teclado una palabra no mayor de 20 caracteres, y la imprima invertida. Use pilas y colas.
- Dado un arreglo constituido de números enteros y que contiene N elementos siendo $N \geq 1$, implemente una solución que diga si la suma de la primera mitad de los enteros del array es igual a la suma de la segunda mitad de los enteros del arreglo.
- Escribir una función recursiva para calcular la altura de un árbol cualquiera.
- Escribir una función no recursiva para calcular la altura de un árbol cualquiera.
- Resolver lo siguiente: Supongamos que tenemos una función *valor* tal que dado un valor de tipo *char* (una letra del alfabeto) devuelve un valor entero asociado a dicho identificador.
- Supongamos también la existencia de un árbol de expresión T cuyos nodos hoja son letras del alfabeto y cuyos nodos interiores son los caracteres $*$, $+$, $-$, $/$.
- Diseñar una función que tome como parámetros un nodo y un árbol binario y devuelva el resultado entero de la evaluación de la expresión representada.
- Implementar una función no recursiva para recorrer un árbol binario en inorden.
- Escribir una función recursiva que encuentre el número de nodos de un árbol binario.
- Realizar un programa que imprima un grafo.
- Construir un programa que determine el número de componentes conexas que posee un grafo cualquiera.
- Resolver lo siguiente: Un grafo no dirigido se dice de Euler si existe un camino Euleriano que incluye a todas sus aristas. Construir una función que dado un grafo no dirigido determine si es de Euler o no lo es.
- Realizar dos cuadros comparativos que reflejen la complejidad en el tiempo y en el espacio de los métodos de:
 - a. Ordenamiento.
 - b. Búsqueda.