|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Programa Educativo: | | | **Ingeniería en Telemática y Sistemas** | | | | | | | | | | Clave: | **PRO4-07-01** | |
|  | | |  | | | | | | | | | |  |  | |
| Nombre de la Asignatura: | | | | **Programación** | | | | | | | | | | | | | |
|  | | | |  | | | | | | | | | | | | | |
| Objetivo General de la Asignatura: | | | | | | | | | | | | | | | | | |
| **Conocer la metodología de programación Orientada a Objetos, Realizar programas simples en el lenguaje JAVA e introducir al estudiante a las buenas prácticas de la programación.** | | | | | | | | | | | | | | | | | |
| Propósito General de la Asignatura: | | | | | | | | | | | | | | | | | |
| **Brindar al estudiante las capacidades base para manejar a un nivel básico-intermedio el lenguaje JAVA y la Programación Orientada a Objetos para que posteriormente sea capaz de continuar con el desarrollo de las habilidades de programación en la carrera.** | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| Ubicación curricular: | | Semestre: | | | | **Cuarto Semestre** | | | | | | | | | | | |
| Antecedente (s): | | | | **Lógica de Programación** | | | | | | | | | | | |
| Consecuente (s): | | | | **Ninguna** | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| Carga curricular: | | Semanal: | | | **4** | | hrs. | Semestral: | | **72** | hrs. | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| Perfil del Alumno: | | | | | | | | | | | | | | | | | |
| **El estudiante deberá contar con una inclinación a la resolución de actividades en casa, trabajo colaborativo, afinidad a la tecnología y a la experimentación con las herramientas que se le brindarán.** | | | | | | | | | | | | | | | | | |
|  | | | | | | | | | | | | | | | | | |
| **Elaboró:** | Ing. Matilde Maldonado | | | | | | | |  | | |  | | |  | | |
| **Revisó:** | Mtro. Ulises Ponce Mendoza | | | | | | | | **Clave de Revisión:** | | | | | | 02-01-2011 | |
| **Autorizó:** | M.C. Cristian Vinicio López del Castillo | | | | | | | |  | | | **Fecha:** | | | 13-01-2011 | | |

**Contenido Temático**

**Unidad I.- El lenguaje java**

Asignatura, Unidad/Tema:

**Familiarizarse con el lenguaje orientado a objetos java y sus herramientas de trabajo.**

Objetivo de la Unidad/Tema:

**10**

Tiempo Estimado hrs.

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * 1. **Introducción a java**   2. **Primer programa en java (formato, comentarios)**   3. **Paquetes en java, tipo de datos, operadores aritméticos, de igualdad, relacionales**   4. **Estructuras de selección simple, doble y múltiple** | * El alumno conocerá las características del lenguaje java, las ventajas de la POO y las herramientas de trabajo * Conocerá la estructura de una aplicación en java, su formato y sus comentarios * Conocerá el funcionamiento de las API´s de java, documentación y operaciones básicas * Conocerá el funcionamiento de las estructuras selectivas | * **El maestro explicará el funcionamiento de este lenguaje, las herramientas con las que cuenta como el IDE, las API’s y la JVM** * **Práctica 1. El alumno se familiarizará con la sintaxis de las aplicaciones en java mediante ejercicios sencillos** * **Práctica 2. Investigar en el sitio de Sun la documentación para java, el IDE de trabajo y el SDK** * **Práctica 3. Escribir aplicaciones utilizando estructuras selectivas** |
| **Criterios de evaluación de la unidad:** las prácticas tendrán un valor del 50%, el conocimiento y manejo del lenguaje con su documentación y creación de aplicaciones sencillas será el otro 50% | | |

**Contenido Temático**

**Unidad II.- Manejo de estructuras repetitivas**

Asignatura, Unidad/Tema:

**El alumno aprenderá el funcionamiento y aplicación de las estructuras repetitivas en el lenguaje java**

Objetivo de la Unidad/Tema:

**10**

Tiempo Estimado hrs.

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * 1. **Estructura de repetición while**   2. **Contadores, acumuladores, operadores unarios, operadores de asignación**   3. **Estructura de repetición do- while**   4. **Estructura de repetición for** | * El alumno aprenderá a crear aplicaciones utilizando estructura repetitiva while * Aprenderá el uso y manejo de contadores y acumuladores * Aprenderá el uso y manejo de la estructura do- while * Aprenderá el uso y manejo de la estructura de repetición for | * **Práctica 1. Elaborar ejercicios utilizando la estructura while** * **Práctica 2. Elaborar ejercicios utilizando contadores y acumuladores** * **Práctica 3. Elaborar ejercicios utilizando do- while** * **Práctica 4. Diseño de aplicaciones utilizando la estructura de repetición for.** |
| **Criterios de evaluación de la unidad:** Los instrumentos de evaluación son: el diseño de un programa elaborando su algoritmo y seudocódigo con las herramientas básicas para la solución de un problema, además de la evaluación diaria de las prácticas y/o ejercicios. | | |

**Contenido Temático**

**Unidad III.- Métodos**

Asignatura, Unidad/Tema:

**Conocer los módulos de programas con java**

Objetivo de la Unidad/Tema:

**6**

Tiempo Estimado hrs.

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * 1. **Introducción.**   2. **Uso de métodos de las API’s de java (por su clase, por el objeto)**   3. **La clase Math**   4. **La definición de un método**   5. **Alcance de las declaraciones**   6. **Recursividad**   7. **Métodos static** | * Conocer el funcionamiento de los métodos y su importancia en la POO. * El alumno aprenderá a manejar métodos de las API’s de java identificando su sintaxis * Los principales métodos de la clase Math * Aprenderá cómo definir un método y hacerlo funcionar * Conocer la injerencia de la declaración de variables * Métodos recursivos * Funcionamiento y uso de los métodos static. | * **El maestro explicará la programación modular, cómo dividir un programa en módulos y su funcionamiento** * **Práctica 1. Elaborar una aplicación utilizando métodos de clases básicas** * **Práctica 2. Diseñar aplicaciones utilizando métodos de la clase Math.** * **Práctica 3. Diseñar métodos que realicen operaciones sencillas y las asignen a variables.** * **El maestro explicará el funcionamiento de las variables locales y los argumentos)** * **Práctica 4. Diseñar ejemplos que apliquen el principio de recursividad con llamadas a métodos.** * **Práctica 5. Diseñar una aplicación que utilice métodos static de una clase existente.** * **Práctica 6. Diseñar métodos static para una clase creada por el programador.** |
| **Criterios de evaluación de la unidad: las prácticas realizadas aportará un 50% de la evaluación total, mientras que la creación y manejo de métodos generará el otro 50%** | | |

**Contenido Temático**

**Unidad IV.- Arreglos**

Asignatura, Unidad/Tema:

**Utilizar arreglos de datos para aplicarlos en la solución de problemas en java**

Objetivo de la Unidad/Tema:

**12**

Tiempo Estimado hrs.

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * 1. **Definición de arreglo**   2. **Declaración y creación de arreglos**   3. **Paso de arreglos a métodos**   4. **Ordenamiento, búsqueda**   5. **Arreglos Multidimensionales** | * El alumno conocerá el manejo dentro de lenguaje java * Conocer el comportamiento de un arreglo * Trabajar con arreglos y métodos * Trabajar con datos dentro de los arreglos y manejarlos * Aprenderá a manipular arreglos de 2 o más dimensiones | * El maestro explicará el concepto de arreglos y su comportamiento en java * Práctica 1. Diseñar ejercicios con manejo de arreglos de datos * Práctica 2. Realizar ejercicios pasando arreglos a métodos y regresando valores * Práctica 3. Realizar ejercicios utilizando métodos con arreglos y procesos de ordenamiento y búsqueda. * Práctica 4. Diseñar ejercicios con arreglos multidimensionales |
| **Criterios de evaluación de la unidad:** El desarrollo de las prácticas tendrán un valor del 50%, la capacidad para resolver un problema utilizando arreglos y métodos contará el 50%. | | |

**Contenido Temático**

**Unidad V.- La clase**

Asignatura, Unidad/Tema:

Objetivo de la Unidad/Tema:

**El alumno aprenderá a crear clases definidas por el programador que ayuden a la solución de problemas dentro de una organización**

Tiempo Estimado hrs.

**8**

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * 1. **Introducción**   2. **La estructura de una clase**   3. **La clase y la aplicación**   4. **Modificadores de acceso**   5. **Paquetes**   6. **Cadenas y caracteres** | * Conocerá el concepto y la función de una clase para crear objetos * Variables de instancia, variables de clase, variables locales * Conocerá la diferencia entre clase y aplicación y su relación al crear un programa * Conocerá el control de la información mediante modificadores de acceso * Uso y manejo de paquetes en java * La clase String | * El maestro explicará el concepto de clase, su funcionamiento dentro de la POO y su estructura general * El maestro expondrá la función de cada uno de los elementos de una clase * Práctica 1. Definir una clase que sirva para generar objetos de esa misma clase. * Práctica 2. Crear aplicaciones que utilicen clases definidas por el programador * Práctica 3. Realizar ejercicios utilizando modificadores de acceso public, private, nulos * Práctica 4. Crear paquete de clases y utilizar clases de paquetes existentes. * Práctica 5. Crear clases utilizando los métodos de la clase String |

|  |
| --- |
| **Criterios de evaluación de la unidad:** Las prácticas tendrán un valor del 50%, las habilidades para crear clases que generen objetos y apliquen comportamientos será el otro 50% |

**Contenido Temático**

**Unidad VI.- Herencia y Polimorfismo**

Asignatura, Unidad/Tema:

Objetivo de la Unidad/Tema:

**El alumno aprenderá el proceso de reutilización de software par el ahorro de tiempo en el desarrollo de programas**

Tiempo Estimado hrs.

**14**

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| * + 1. **Herencia**     2. **Modificadores de acceso protected**     3. **Constructores y finalizadotes en las clases derivadas**     4. **Polimorfismo**     5. **Uso de referencias a superclase con variables tipo subclase**     6. **Llamadas a métodos de subclase mediante variables de superclase**     7. **Excepciones** | * Entenderá el concepto de herencia, clase base y clase derivada * Uso de modificadores de acceso protected * la instrucción super, finalize * El concepto e polimorfismo y su utilización en ingeniería de software * Referencias a objetos * manejo de métodos * manejo de excepciones para validación de clases | * Realizar esquemáticamente el concepto de herencia diferenciando entre clases y superclases * Práctica 1. Creación de clase base y clase derivada (extends) * Práctica 2. Ejercicio práctico con jerarquía de clases * El maestro explicará con ejemplos la diferencia entre herencia y polimorfismo * Práctica 3. Creación de ejercicios utilizando polimorfismo * Práctica 4. Utilización de métodos de la subclase * Práctica 5. Realización de clases manejando excepciones. |

|  |
| --- |
| **Criterios de evaluación de la unidad:** Las prácticas tendrán un valor del 50%, las habilidades para manejo de herencia y polimorfismo dominando el concepto tendrá un valor del 50% |

**Contenido Temático**

**Unidad VII.- la interfaz gráfica de usuario**

Asignatura, Unidad/Tema:

Objetivo de la Unidad/Tema:

**El alumno aprenderá a crear sus propios diseños con ventanas, botones y demás objetos con los que cuenta las API’s de java para diseñar sus programas**

Tiempo Estimado hrs.

**12**

| **Temática** | **Aprendizaje** | **Estrategias** |
| --- | --- | --- |
| 1. **Los componentes Swing** 2. **Etiquetas, botones, listas** 3. **Manejo de eventos** 4. **Cuadros de texto** | * 1. Conocerá las clases que crean los componentes de la interfaz gráfica de usuario   2. Aprenderá como crear botones, etiquetas, listas, etc.   3. Aprenderá cómo darle funcionalidad a botones   4. Aprenderá a crear sus cuadros de texto | * El maestro explicará las ventajas de los componentes GUI del paquete javax.swing * Práctica 1. Diseño de una clase con objetos gráficos * Práctica 2. Hacer funcionar los botones de una clase(aplicación) creada por el programador * Práctica 3. Agregarle cuadros de texto a su aplicación. |

|  |
| --- |
| **Criterios de evaluación de la unidad:** Las prácticas tendrán un valor del 50%, las habilidades para crear clases utilizando interfaz gráfica de usuario valdrá el otro 50% |

**Criterios de Evaluación y Acreditación:**

|  |
| --- |
| **Evaluación:** |
| Las actividades de evaluación deberán ser continuas e integrales, centrándose principalmente en tres aspectos a saber: a) Habilidades.- Son las destrezas manuales, procedimentales y cognitivas que el alumno puede evidenciar al momento de la resolución de problemas. (el saber hacer); b) Actitudes.- Son respuestas del alumno ante las diversas situaciones sociales que se le presentan (el saber ser); c) Conocimientos: Es el saber teórico-conceptual que se puede incrementar. (el saber).  Como herramientas de evaluación de las habilidades y conocimientos, se sugieren las siguientes: elaboración de un ensayo, exposiciones, mapas conceptuales, socio-dramas, resolución de problemas, estudios de caso, avances de proyectos/investigación,  reportes de lectura, prácticas de laboratorio y taller, ejercicios de evaluación, prácticas de campo, portafolio de evidencias, discusión analítica, participaciones significativas en clase, exámenes ó evaluación oral/escrita no calendarizados(as).  (No se permitirá ningún tipo de actividad de reposición)  Como herramientas de evaluación de las actitudes, se sugieren: a) bitácoras de puntualidad, entrega oportuna de trabajos y proactividad; b) autoevaluación comentada; c) evidencia de participación en su comunidad de aprendizaje. |
|  |
| **Acreditación:** |
| Para acreditar el curso el alumno deberá cumplir con el 90% de las asistencias regulares del curso, además deberá presentar una evaluación mínima aprobatoria (70 ptos.) en todos y cada uno de los aspectos a evaluar, si faltase uno de ellos, será sujeto de no acreditación. |

**Bibliografía:**

|  |
| --- |
| Básica: |
| * Deitel/ Deitel, Java cómo programar, Prentice Hall * Luis Joyanes Aguilar , Java 2 El lenguaje de programación , Mc Graw- Hill. * Francisco Javier Ceballos, El lenguaje de programación java, Alfaomega |
| Complementaria: |
| * *Bruce Eckel, Piensa en java, Prentice Hall* |

|  |  |  |
| --- | --- | --- |
| **Modificaciones:** | | |
| **Revisión** | **Modificación** | **Fecha** |
| 00-01-2009 | * Base | DD-MM-AAAA |
| 01-01-2010 | * Revisión | 10-01-2010 |
| 02-01-2011 | * Revisión | 13-01-2011 |