 <b>IKIAM</b> UNIVERSIDAD REGIONAL AMAZÓNICA	<b>UNIVERSIDAD REGIONAL AMAZÓNICA IKIAM</b>	
	<b>Carreras:</b> Ingeniería en Ciencias del Agua Ingeniería en Ecosistemas	<b>Syllabus de asignatura</b> <b>Segundo Semestre</b>

## 1. INFORMACIÓN GENERAL

<b>Asignatura:</b>	<b>Matemáticas II - Algoritmia y Programación</b>
<b>Unidad Curricular</b>	<b>Básica</b>
<b>Nivel</b>	<b>Segundo Semestre</b>
<b>Campos de formación</b>	<b>Fundamentos teóricos</b>
<b>Pre-requisitos</b>	<b>Matemáticas I</b>
<b>Co-requisitos</b>	<b>Ninguno</b>

## 2. DESCRIPCIÓN DE LA ASIGNATURA

El curso trata del uso eficiente de las computadoras para resolver problemas reales de cómputo, así como de gestión de datos.

## 3. OBJETIVOS

Que el o la estudiante adquiera la competencia de desarrollar sus propios programas informáticos para solucionar cuantos problemas numéricos pueda encontrarse en su carrera profesional.

## 4. CONTENIDOS

<b>Algoritmia y Programación (80 horas)</b>	<p>1. Introducción a la programación imperativa (12 horas): Se mostrarán los conceptos básicos para el modelo clásico de la programación de computadoras. Para ello se introducirán las herramientas de las cuales se dispone para la elaboración de programas informáticos.</p> <p>2. Variables (2 horas): Tanto desde un punto de vista material, el espacio correspondiente de memoria ocupado por una variable en cualquier lenguaje de programación, como desde el punto de vista conceptual de lo que significa una variable, su dominio, y su incidencia en la consecución de resultados a problemas concretos, se tratará en esta parte introductoria del curso.</p>
---	--

3. Estructuras fundamentales del código imperativo (2 horas):

Desde que a finales del siglo XX se introdujo la programación estructurada como la mejor opción para realizar código comprensible, las estructuras de cualquier lenguaje de programación han sido aceptadas por consenso en todos los lenguajes imperativos que se usan en la actualidad.

4. Lenguajes de programación (2 horas):

Antes de introducirnos intensamente en la elaboración de soluciones para los problemas que puedan presentarse en cualquier disciplina científica, ya sea biología, química, física o geología, es conveniente observar el estado de la situación desde una perspectiva global, para comprender en que ámbito de todo el espectro se va a situar nuestra especialización.

5. Entornos integrados de desarrollo IDE (2 horas):

Se presentarán en esta parte del curso las herramientas que cualquier estudiante debe manejar con soltura con el propósito de convertir la computadora en su forma de resolver cualquier tipo de problema que se le plantee. Los entornos integrados de desarrollo son los programas informáticos que cualquier desarrollador debe dominar.

6. El entorno netbeans (4 horas):

Alguno de los entornos integrados de desarrollo debe ser usado con fluidez por una persona capaz de desarrollar programas informáticos. El entorno netbeans es uno de los IDEs de código abierto con más prestigio de los que se encuentran actualmente en el mercado.

7. El lenguaje JAVA (10 horas):

Cada uno de los lenguajes de programación que se usan en la actualidad se asocia a una especialización que lo caracteriza y es el motivo por el cual fue diseñado. El lenguaje de programación JAVA es sin duda el lenguaje de propósito general más utilizado en la actualidad. Por esta razón es conveniente que los y las estudiantes sean capaces de resolver los problemas planteados usando este lenguaje, que encaja perfectamente en el entorno netbeans.

8. Estructuras de datos (2 horas):

Conceptos como modularidad y empaquetamiento son fundamentales para la portabilidad del código. Las estructuras de datos son la forma más emblemática de reunir conjuntos heterogéneos de datos bajo un único concepto y como tales, es

conveniente adquirir soltura en su diseño e implementación.

9. Objetos (2 horas):

Los objetos como consecuencia de los elementos primigenios que fueron las variables en la parte introductoria de la asignatura, y una vez evolucionados hasta las estructuras de datos, se convierten en el elemento central de la nueva programación nacida en los años noventa. Es importante comprender su definición para materializarlos en todos los niveles de abstracción que contempla la programación de computadoras.

10. Programación orientada a objetos (16 horas):

Esta técnica de programación es sin duda una de las técnicas de programación informática que más ha cuajado en todos los lenguajes de programación. Y por ello es deber incuestionable comprender y razonar a partir de ella.

11. Clases (6 horas):

Las clases son la materialización de la programación orientada a objetos, y una extensión evolutiva de lo que en un principio fueron los tipos estándar de variables. La relación entre variable y objeto debe ser comprendida de forma análoga a la que puede establecerse entre tipos y clases.

12. Herencia (6 horas):


Una extensión de gran impacto que se deriva del uso de las clases sustituyendo los tipos de datos inmediatos es la herencia. En este curso de programación de computadoras se considera el dominio de las herencias y los polimorfismos como algo fundamental en el desarrollo de aplicaciones informáticas, así como en la modelización de datos.

13. Interfaces gráficas de usuario (4 horas)

Puesto que en los distintos protocolos, red, escritorio, tienen la característica común de usar interfaces gráficas, se promocionará la capacidad de diseñar e implementar tales interfaces. Eso, a su vez, conducirá a la programación orientada por eventos.

14. Bases de datos (10 horas):

Es necesario disponer de un mínimo conocimiento de las bases de datos relacionales para poder reunir proyectos de origen diverso asentando los métodos estándar de interacción entre el código y los datos que maneja.

	<b>UNIVERSIDAD REGIONAL AMAZÓNICA IKIAM</b>	
	<b>Carrera en Ingeniería en Ciencias del Agua</b>	<b>Syllabus de asignatura Primer Semestre</b>

## 5. EVALUACIÓN

La evaluación de este curso se integra de tres partes la conjunción de las cuáles servirá para obtener la calificación final de la asignatura:

1. Habrá una parte de teoría en un examen final (F)
2. Una parte formada por ejercicios que el o la estudiante deberá resolver fuera de horas de clase. (E)
3. Un trabajo práctico que incluirá el examen parcial, que los estudiantes deberán resolver totalmente fuera de horas de clase. también. (P)

La calificación final se obtendrá con el promedio de las tres partes.

$$\text{NOTA} = 0.33 \text{ F} + 0.33 \text{ E} + 0.33 \text{ P}$$

## 6. BIBLIOGRAFÍA

- Thinking in Java. Bruce Eckel, 4th edition, Prentice Hall.
- Programación en Java. Jesús S. Allende et al, 3ª edición, McGraw-Hill, 2009.
- Joyanes, L. Fundamentos de programación : algoritmos, estructuras de datos y objetos. McGraw-Hill : Madrid, 2003.
- Meyer, Bertrand. Construcción de software orientado a objetos. Madrid : Prentice Hall, 1998.
- Budd, Timothy. An Introduction to object-oriented programming. Boston [Mass.] [etc.] : Addison-Wesley, 2002.