

LABORATORIO DE COMPUTACIÓN I

Lic. Cristina Domizio – Ing. Gerardo Magni

Planificación

FUNDAMENTACIÓN DE LA MATERIA DENTRO DEL PLAN DE ESTUDIOS

La estructura curricular del espacio bajo análisis dispone de 6 horas-cátedra semanales. Se propone un aula-taller que básicamente persigue el objetivo de incrementar en el alumno su natural habilidad para utilizar el pensamiento elaborado lógicamente a fin de resolver problemas a través de la descomposición de los mismos, en subproblemas más simples (diseño descendente), estableciendo la relación entre el todo y las partes (enfoque sistémico), considerando sólo los aspectos relevantes (abstracción), mediante el uso de herramientas fundamentales representativas de los procesos deductivos asociados a la programación, integrando la sintaxis elemental de un lenguaje de programación en Laboratorio. A su vez se intenta modelar un ambiente creativo de niveles crecientes de realismo respecto del mundo laboral pertinente. Esto nos lleva a establecer la metodología de carpeta abierta y uso de documentación en soportes y en línea.

En definitiva el espacio curricular es uno de los más propicios para lograr “El desarrollo del pensamiento reflexivo, del juicio crítico, de la creatividad y de la capacitación para el auto-aprendizaje”¹, y a la vez, propiciar el derecho del alumno a recibir “formación científica, profesional, humanística y técnica en el más alto nivel, contribuir a la preservación de la cultura nacional, promover la generación y desarrollo del conocimiento en todas sus formas, y desarrollar las actitudes y valores que requiere la formación de personas responsables, con conciencia ética y solidaria, reflexivas, críticas, capaces de mejorar la calidad de vida, consolidar el respeto al medio ambiente, a las instituciones de la República y a la vigencia del orden democrático”².

Es obvio, luego, el rol de auxiliar y complementaria de esta asignatura, posibilitando un enfoque más práctico y orientado al uso de herramientas presentes en un entorno integrado de desarrollo (IDE por sus siglas en inglés).

¹Ley 6970, Título II: Principios y Fines de la Educación, Cap II: Fines de la Educación, Art. 6°, inc. C.

²Ley 24521, Título II: Capítulo 1: De los Fines y Objetivos, Art. 3°.

Finalmente, debemos destacar que, además de ser la programación uno de los campos más específicos, aplicables y solicitados dentro del mundo laboral, es a la vez, claramente influyente en la corroboración vocacional del educando. Así el mismo disfrutará (o padecerá) intensamente el exigente cursado de esta asignatura.

OBJETIVOS DE LA ASIGNATURA

La asignatura persigue diferentes objetivos a alcanzar por el alumno. En principio, los contenidos a impartir pueden clasificarse de la siguiente manera:

Contenidos procedimentales

Se aspira a que, a lo largo del desarrollo del espacio curricular se adquieran los siguientes:

Soluciones creativas a problemas: se pretende que cualquiera sea el problema enfrentado exista al menos una metodología de solución del mismo, utilizando las herramientas que brinda el ambiente de desarrollo.

Adquisición de principios de buen diseño de software: éstos, cuando se utilizan, minimizan los riesgos de error y el esfuerzo requerido para alcanzar una solución a un problema.

Planificación de tareas complejas: el desarrollo de software vincula múltiples elementos en la consecución de una solución a un problema dado. Esto requiere de planificación, es decir, de la descomposición de tareas en subtareas más sencillas.

Representación y comunicación de ideas: para que la idea sea comunicable debe estar bien formada y justificada. Para comunicarla a otros es además necesario organizar contenidos en forma lógica, respetando las precedencias y, en general, una complejidad creciente. Esta habilidad implica además el uso de herramientas adecuadas para tal fin.

Contenidos actitudinales

Desarrollo personal: confianza en sí mismo, en la resolución de problemas, perseverancia en la consecución de objetivos, creatividad en la búsqueda de soluciones y apasionamiento por el trabajo autónomo y en grupo.

Desarrollo socio – comunitario: disposición para cooperar en trabajos grupales y respeto por las opiniones diferentes.

Desarrollo de la curiosidad investigativa: disposición positiva para la indagación, atención hacia los problemas implícitos en la resolución de un problema, desarrollo de la visión crítica y autocrítica de problemas y soluciones, valoración de los aportes de diferentes campos del

conocimiento, en la comprensión de los enunciados de problemas. Cuidado y uso racional de recursos.

Desarrollo de la comunicación y de la expresión: valoración del uso del vocabulario e incorporación del mismo en la comunicación de resultados. Cuidado en la elaboración de la argumentación propia. Análisis reflexivo y crítico de cualquier argumentación. Aprecio por la claridad, calidad y pertinencia en la presentación de resultados.

METODOLOGÍA DE ENSEÑANZA

El dictado de la cátedra se deberá realizar en el laboratorio que posee la universidad para tal fin, contando con la tecnología correspondiente a la última máquina virtual disponible para Java como así también la última versión del Framework de desarrollo de Netbeans.

El procedimiento de enseñanza se realizará del siguiente modo:

Cada clase se comenzará con la reseña teórica correspondiente a la planificación.

Posteriormente se explicará en clase un ejercicio que represente los contenidos enseñados en los temas teóricos de tal manera que se afirme mediante un ejemplo el algoritmo y las instrucciones de código que en él intervienen.

El alumno deberá completar en forma individual y como tarea complementaria fuera del horario de cursado otros ejercicios de similares características al impartido en clase.

METODOLOGÍA DE EVALUACIÓN, INSTANCIAS DE RECUPERACIÓN Y RÉGIMEN DE PROMOCIÓN.

Se implementará una metodología de evaluación continua de cada alumno. Dichas instancias de evaluación serán de carácter práctico y teórico.

En el transcurso del semestre, se desarrollarán trabajos prácticos individuales que deberán ser presentados en tiempo y forma para su corrección por cargo del docente.

Se tomarán dos (2) evaluaciones parciales y una evaluación Global, todas teórico-prácticas y recuperables.

Los alumnos obtendrán la Aprobación Directa con 6 o más (60% o más), Aprobación No Directa con 4 ó 5 (40% a 59%) y No Aprobación con 1, 2 y 3 (1 a 39).

También deben considerar que aprobado es con 60% ó más para los exámenes finales.

Todos los que no alcancen del 60% en ambos parciales, pueden recuperar sólo una vez, cada examen desaprobado, para alcanzar la Aprobación Directa o la No Directa. La directa será posible si rinde examen global.

El alumno que en ambos parciales y/o recuperatorios obtenga calificaciones menores al 60%, no podrá rendir el examen Global.

Si el alumno rinde global, éste dará la nota final de la materia.

La escala de porcentajes que se aplicará es la siguiente:

Nota	Escala Porcentual
1	1 a 12 %
2	13 a 24 %
3	25 a 39 %
4	40 a 47 %
5	48 a 59 %
6	60 a 64 %
7	65 a 74 %
8	75 a 84 %
9	85 a 94 %
10	95 a 100 %

RECURSOS DIDÁCTICOS A UTILIZAR COMO APOYO A LA ENSEÑANZA

Los recursos didácticos se apoyan en el proyector multimedia que poseen los laboratorios, videos Tutoriales, apuntes y ejercicios desarrollados en el lenguaje de Java por el docente a cargo.

PROYECTO DIDÁCTICO DE PRUEBA COMO APOYO AL APRENDIZAJE Y FIRMEZA DE CONOCIMIENTOS DE LOS ALUMNOS

- a) Deseamos que los alumnos aseguren su aprendizaje y para eso utilizaremos como herramienta, el trabajo grupal compitiendo con otros grupos. Creemos que esto puede ayudarlos en la investigación y en el desarrollo de metodologías para lograr los objetivos, buscando ser los ganadores.
- b) El objetivo será una propuesta en común, dada por los profesores de la cátedra, la cual indicará los objetivos a cumplir y principalmente, sus características serán el uso de todos los conocimientos impartidos en la cátedra durante el semestre. La utilización de tales herramientas, implicará un puntaje específico.
- c) La participación en la competencia se deberá indicar al iniciar el semestre y será opcional, es decir, que no será obligatoria para el alumno. Sin embargo, si el grupo que participa gana, tendrá un beneficio en la calificación final de la cátedra.

TRABAJOS DE CAMPO, VISITAS A EMPRESAS.

Se coordinará una invitación a una Software Factory para que la misma comunique la importancia del rol del programador dentro de la organización justificando sus funciones.

ARTICULACIÓN HORIZONTAL Y VERTICAL CON OTRAS MATERIAS

En cuanto a la articulación horizontal:

Planificación de la Cátedra Laboratorio de computación I
Lic. Cristina Domizio – Ing. Gerardo Magni

La cátedra de Laboratorio I debe articularse en su cronograma con la cátedra de Programación I ya que se codificará en Java los algoritmos planteados en la misma.

En cuanto a la articulación vertical:

La cátedra de Laboratorio I funda las bases en cuanto a los primeros pasos de programación en Java y el manejo del Framework para las cátedras de Laboratorio II y Programación II del siguiente semestre.

CRONOGRAMA ESTIMADO DE CLASES, ACTIVIDADES Y EVALUACIONES

Semana 1:

Introducción a la programación –Introducción a Java- Máquina virtual – Editores – Intérpretes–
Compiladores. Programa fuente – Bytecode - Package
Tipos de datos – Definición de variables

Semana 2:

Introducción a la Ide de programación Netbeans – Conceptos de proyecto, paquetes
Clases principal, método main - Estructura de un programa en java – Estructuras secuenciales - Introducción a métodos estáticos – Ejercitación.

Semana 3:

Sentencias condicionales – Condición simple – Condición compuesta - Anidamiento

Semana 4:

Iteraciones – Ciclo for – While –

Semana 5:

Iteraciones - Do while

Semana 6:

Corte de control simple – Corte de control compuesto

Semana 7:

Primer parcial y entrega de la carpeta de trabajos prácticos. Elaboración de ejercicios alternativos

Semana 8:

Introducción a arreglos unidimensionales – Declaración – Inicialización – Carga. Ejercitación

Semana 9 :

Carga. Ejercitación de arreglos unidimensionales

Semana 10:

Introducción a matrices . – Declaración – inicialización – Carga. Ejercitación
Semana 11:
Métodos de ordenamiento – Bubble sort – Selection sort – Insertion sort – Shell
sort – Heap sort – merge sort – Quick sort .
Semana 12:
Ejercitación con métodos de ordenamiento y algoritmos de búsquedas..
Semana 13:
Segundo Parcial – Entrega de carpeta de trabajos prácticos.
Semana 14:
Examen Global - Recuperación del primer y segundo parcial.

BIBLIOGRAFÍA, GUÍAS DE ESTUDIO, PUBLICACIONES VINCULADAS CON LA MATERIA.

Obligatoria :

Java How to Program 7/e. Deitel & Deitel. Ed. Prentice Hall.
JAVA 2: Curso de Programación. Fco. Javier Ceballos Editorial
Alfaomega- Rama.
Apuntes del docente.

Complementaria :

Manual de JAVA: Patrick Naughtn – MC Graw Hill
Manual Avanzado de Java : F Lima Diaz – Anaya
Introducción a la Programación Orientada a Objetos con Java – Thomas
Wu – Editorial Mc Graw Hill
Pensando en JAVA: Bruce Eckell
La Biblia de Java: Bruce Eckell
Manual de JAVA: Patrick Naughtn – MC Graw Hill

Lugar y fecha: Mendoza, Febrero de 2019	COORDINADORES DE CÁTEDRA Apellido y nombre: Lic. Cristina Domizio Lic Gerardo Magni –