

Facultad de Ingeniería y Ciencias Agropecuarias
Ingeniería en Sistemas de Computación e Informática
ACI325 - PROGRAMACION
Período académico 2017-2

1. Identificación

Número de sesiones: **64**

Número total de hora de aprendizaje: **160**

Créditos – malla actual: **64**

Profesor: **Washington Oswaldo Pérez Argudo**

Correo electrónico del docente (Udlanet): **w.perez@udlanet.ec**

Director: **Marco Antonio Galarza Castillo**

Campus: **Queri**

Pre-requisito (PRODUCCION): **ACI274 TECNOL. INFORMACION**

Co-requisito: **N/A**

Pre-requisito (SONIDO): **ACI356 Computación Aplicada**

Co-requisito: **IES542 SIM. SIT. ACUST.**

Paralelo: **2**

Tipo de asignatura:

Optativa	
Obligatoria	X
Práctica	

Organización curricular:

Unidad 1: Formación Básica	X
Unidad 2: Formación Profesional	
Unidad 3: Titulación	

Campo de formación:

Campo de formación				
Fundamentos teóricos	Praxis profesional	Epistemología y metodología de la investigación	Integración de saberes, contextos y cultura	Comunicación y lenguajes
	X			

2. Descripción del curso

Debemos tener en cuenta para esta materia, que el alumno no tiene conocimiento alguno de la misma, ni experiencias laborales previas en programación. Las clases serán teórico práctico, en donde los estudiantes desarrollarán destrezas y habilidades básicas en el campo de la programación estructurada de sistemas, usando inicialmente para esto flujos y luego introduciéndoles en la programación visual promoviendo la práctica de los valores y la ética profesional.

3. Objetivo del curso

Comprender los conceptos de la programación estructurada en la construcción de programas utilizando un lenguaje de programación visual para la solución de problemas y considerando siempre las buenas prácticas de programación de aplicaciones.

4. Resultados de aprendizaje deseados al finalizar el curso

Se enuncian en la columna izquierda, de manera explícita y desde la perspectiva del aprendizaje, a partir de la pregunta: ¿Cuáles son los resultados de aprendizaje deseados al finalizar este curso? Deben ser claros, evidenciables y evaluables. En la columna 2 a la derecha se indica a qué RdA del perfil de egreso de la carrera aportan para su desarrollo; y en la columna 3 se indica el nivel de desarrollo del RdA de carrera dentro de la malla curricular.

Resultados de aprendizaje	RdA perfil de egreso de Carrera	Nivel de dominio (carrera)
1. Describe los conceptos básicos de la programación de aplicaciones informáticas.	PRODUCCIÓN 1. Piensa lógicamente para lograr la resolución de programas estructurando y administrando los flujos de trabajo de una forma ordenada, secuencial y lógica	Inicial__X__ M____ F____
2. Diseña aplicaciones informáticas para la resolución de problemas computacionales simples.	PRODUCCIÓN 2. Generar soluciones mediante la optimización de procesos para resolver problemas desde una postura fundamentada en un contexto determinado para articular soluciones basadas en argumentos válidos tecnológicos y actuales.	I____ Medio__X__ F____
3.- Elabora programas informáticos utilizando lenguajes de tercera y cuarta generación	PRODUCCIÓN 3. Aplicar herramientas tecnológicas que ayuden a la organización industrial y simulación por computador logrando optimizar los procesos industriales.	I____ Medio__X__ F____

5. Sistema de evaluación

De acuerdo al Modelo Educativo de la UDLA la evaluación busca evidenciar el logro de los resultados de aprendizaje (RdA) enunciados en cada carrera y asignatura, a través de mecanismos de evaluación (MdE). Por lo tanto la evaluación debe ser continua, formativa y sumativa. La UDLA estipula la siguiente distribución porcentual para los reportes de evaluaciones previstas en cada semestre de acuerdo al calendario académico:

Reporte de progreso 1	35%
Ejercicios y talleres en clase	10%,
Trabajos Autónomos (Tareas)	10%,
Examen, teórico práctico	15%.
Reporte de progreso 2	35%
Ejercicios y talleres en clase	10%,
Trabajos Autónomos (Tareas)	10%,
Examen, teórico práctico	15%,
Evaluación final	30%
Ejercicios y talleres en clase	5%,
Examen final	10%,
Proyecto Final	15%,

Asistencia: Es obligatorio tomar asistencia en cada sesión de clase.

Al finalizar el curso habrá un examen de recuperación para los estudiantes que, habiendo cumplido con más del 80% de asistencia presencial a clases, deseen reemplazar la nota de un examen anterior (ningún otro tipo de evaluación). Este examen debe integrar todos los conocimientos estudiados durante el periodo académico, por lo que será de alta exigencia y el estudiante necesitará prepararse con rigurosidad. La nota de este examen reemplazará a la del examen que sustituye. Recordar que para rendir el EXAMEN DE RECUPERACIÓN, es requisito que el estudiante haya asistido por lo menos al 80% del total de las sesiones programadas de la materia. No se podrá sustituir la nota de un examen previo en el que el estudiante haya sido sancionado por una falta grave, como copia o deshonestidad académica.

El examen final constará de dos partes fundamentales:

- ✓ Entrega de documentación de todos los pseudocódigos, diagramas de flujo, y soluciones a problemas, tanto diseño, código fuente y proyecto informático. Esta información será entregada en formato digital (PDF). Como característica fundamental, cada una de las evidencias presentadas, deberá tener la fecha y firma de cuando fue realizada. Esta parte del trabajo final, corresponderá al 40% de la nota final. La rúbrica que se utilizará para este efecto es la siguiente:

Valoración de resultados			
A valorar	Óptimos	Suficientes	No satisfactorios
	De 8 a 9.99	De 5 a 7.99	Menos de 5
Tiempos y condiciones	El trabajo se entrega a tiempo y en el formato previamente establecido	El trabajo se presenta fuera de las fechas establecidas y/o no concuerda con los formatos definidos	No presenta el trabajo o solicita aplazamiento para su entrega
Orden y secuencialidad	El trabajo mantiene un orden lógico y supeditado a la misma lógica de avance de la materia	Si bien el trabajo contiene todo lo solicitado, el orden lógico varía dificultando su entendimiento	El desorden impera en el trabajo presentado, no se encuentra una línea conductual clara
Partes constitutivas	El trabajo muestra todos los	Faltan una parte solicitada	Falta más de una parte de los

	avances desarrollados, desde pseudocódigo, pasando por programación y terminando en la documentación de los aplicativos	(pseudocódigo, diagramas de flujo o documentación de aplicativos)	documentación solicitados
Evidencia de trabajo continuo	Se evidencia claramente que el estudiante ha ido documentando sistemáticamente sus trabajos	Se nota premura e improvisación por presentar el trabajo	Los documentos presentados no tienen ni fecha ni firma que evidencian un trabajo sistémico
Valores agregados propios	Aporta con carátulas, índices, notas a pie de página, etc.	Se limita a entregar el trabajo con una carátula inicial	No existe ningún aporte propio

- ✓ Desarrollo de un aplicativo orientado al manejo de archivos o al tratamiento de gráficos. La decisión final sobre cuál de los dos temas tratar, la tendrá el mismo estudiante, tratando de respetar su tenencia y mejores capacidades. Este trabajo tendrá que realizarlo de manera presencial y tendrá un aporte del 60% de la nota final. La rúbrica que se utilizará es la siguiente:

Valoración de resultados			
A valorar	Óptimos	Suficientes	No satisfactorios
	De 8 a 9,99	De 5 a 7,99	Menos de 5
El resultado final	Se alcanza y se superan los resultados esperados	Se alcanzan los resultados esperados con cierta complejidad	No se alcanzan los resultados esperados
El orden	La solución planteada se muestra totalmente ordenada, lógica y con utilización adecuada de las buenas prácticas de programación	Si bien la solución presentada cumple con lo requerido, no existe un orden adecuado de formas, objetos y código fuente.	No se evidencia orden en codificación, objetos y estandarización de formas es difícil seguir la secuencia de los programas.
La documentación	Se establecen datos como fechas, horas, objetivo de cada programa, autor, archivos de entrada y salida y se comentan las líneas de código adecuada y claramente	Existe auto-documentación, la misma que es suficiente y adecuada, aunque falta la explicación de varias líneas de código que podrían colaborar con la claridad y el entendimiento de la solución	Los comentarios al código fuente o no existen o son demasiado básicos, contribuyendo a la complejidad y agregando dificultad en el entendimiento de los programas
El manejo de la herramienta de programación	Se maneja adecuadamente las interfaces y se evidencia soltura y confianza en el manejo del entorno de la herramienta de programación	Se conocen el ambiente básico de la herramienta, faltando conocimiento del sitio y ubicación de alguno de sus elementos	No se conoce el ambiente en donde se programa, la movilidad entre una herramienta y otra, entre un objeto y otro, y entre una forma y otra. Demuestra desconocimiento y falta de práctica
Interacción con la red informática	Habilita y deshabilita adecuadamente el ambiente de red entre discos duros, discos virtuales y dispositivos externos	Conoce el manejo básico de la red de almacenamiento de datos	No puede grabar, recuperar, encontrar, los archivos que se requieren para iniciar, proseguir y concluir su trabajo
Los aportes personales	Se evidencia aportes de buenas prácticas de programación, utilización de colores y ubicación de objetos, que en las clases habituales no fueron aportes	Se limita a seguir las indicaciones dadas por el docente, las cumple bien, a cabalidad y en forma completa	Sigue instrucciones, aun cuando regularmente hay que recordárselas, sin poder cumplirlas, al final de cuentas

	del docente		
El proceso de innovación y creatividad	Es parte de la solución presentada con aportes que agregan valor al producto final	Se evidencian varios aportes personales	No se evidencia aporte personal alguno

6. Metodología del curso y de mecanismos de evaluación. (Docente)

Respecto de la metodología a utilizar para el desarrollo del aprendizaje de esta materia lo vamos a enfocar respondiendo a las siguientes preguntas:

¿Qué sabemos, como docentes, de cómo nuestros estudiantes llegan a obtener conocimiento y hasta qué punto usted aplica lo que sabe en su práctica diaria?

Es importante poner en práctica el aprendizaje significativo, porque de esta forma podemos determinar el conocimiento que poseen los estudiantes y con esto podemos trazar una hoja de ruta respecto de cómo se afrontará el desarrollo del conocimiento

¿De todas las cosas que nuestros estudiantes puedan llegar a conocer cuáles son las que necesitan aprender ahora y por qué?

Dentro del conocimiento que los estudiantes deben adquirir, lo más importante en esta materia es que el estudiante logre adquirir un pensamiento analítico, que le permita generar varios caminos para la solución de problemas; que piense críticamente y demuestre su capacidad para analizar información y hechos, evaluar ideas y alternativas propias y de otros desde diferentes puntos de vista

¿Sabiendo lo relativo a las preguntas anteriores, como se organiza la experiencia de aprendizaje para responder a las necesidades de sus estudiantes?

El aprendizaje, debe ser positiva que le represente al estudiante una experiencia que le ayude aplicar el razonamiento lógico y matemático para resolver retos complejos, así como problemas cotidianos

Teniendo en cuenta las preguntas anteriores la metodología aplicar en esta materia es la constructivista esta se enfoca en la construcción del conocimiento a través de actividades basadas en experiencias ricas en contexto. El constructivismo ofrece un nuevo paradigma para esta nueva era de información motivado por las nuevas tecnologías que han surgido en los últimos años. Con la llegada de estas tecnologías como las web 2.0, los estudiantes no sólo tienen a su alcance el acceso a un mundo de información ilimitada de manera instantánea, sino que también se les ofrece la posibilidad de controlar ellos mismos la dirección de su propio aprendizaje.

Debemos tener en cuenta el vínculo entre el uso efectivo de las nuevas tecnologías (Aulas Virtuales UDLA Moodle) y la teoría constructivista, explorando cómo las tecnologías de la información aportan aplicaciones que al ser utilizadas en el proceso de aprendizaje, dan como resultado una experiencia de aprendizaje excepcional para el individuo en la construcción de su conocimiento, describiendo los conceptos básicos de la programación de aplicaciones informáticas.

Cambiar el esquema tradicional del aula, donde el papel y el lápiz tienen el protagonismo principal, y establecer un nuevo estilo en el que se encuentren presentes las mismas herramientas pero añadiéndoles las aplicaciones de las nuevas tecnologías, aporta una nueva manera de aprender, que crea en los estudiantes una experiencia única para la construcción de su conocimiento. El punto central de esta investigación es analizar cómo las nuevas tecnologías como herramientas constructivistas intervienen en el proceso de aprendizaje de las personas; es por esta razón que utilizaremos los siguientes componentes: (Foros Virtuales, Evaluaciones, Carga de archivos, Desarrollo de Blogs)

El estudiante deberá ingresar mínimo 1 vez luego de cada clase presencial, para responder o desarrollar la actividad respectiva Las participaciones serán evaluadas de acuerdo a la rúbrica respectiva, logrando que el estudiante diseñe aplicaciones informáticas para la resolución de problemas computacionales simples.

Las metodologías y mecanismos de evaluación deben explicarse en los siguientes escenarios de aprendizaje:

6.1. Escenario de aprendizaje presencial.

Aprendizaje Colaborativo, desarrollo de proyectos que general interés en los estudiantes logrando que los participantes aprendan haciendo, poniendo en práctica conocimientos y destrezas para obtener un producto nuevo; enseñanza unidireccional con el uso de clases magistrales en algunos típicos .

6.2. Escenario de aprendizaje virtual.

Foros y debates que permitan analizar las mejores prácticas para lograr de esta forma, compartir ideas, opiniones, y discutir temas de actualidad que permitan afianzar el conocimiento y destrezas del estudiante. Además realizar prácticas con la finalidad de reforzar el conocimiento impartido en cada clase para de estar forma utilizar una situación auténtica los conocimientos y destrezas adquiridas en clase.

6.3. Escenario de aprendizaje autónomo.

Proyectos y exposiciones que permitan aplicar conocimientos y destrezas para crear ideas nuevas y compartirlas; además enseñanza enfocada en el estudiante, usando portafolio de ejercicios.

7. Temas y subtemas del curso

RdA	Tema	Subtemas	
Describe los conceptos básicos de la programación de aplicaciones informáticas.	Conceptos Básicos de Programación	1.1	Los lenguajes de programación actuales
		1.2	Algoritmo
		1.3	Variables y constantes
		1.4	Tipos de datos

		1.5	Constante
		1.6	Operadores
		1.7	Funciones
	Programación Estructurada	2.1	Conceptos de programación estructurada
		2.2	Programación mediante saltos y rutinas
Diseña aplicaciones informáticas para la resolución de problemas computacionales simples.	Programación Orientada a Objetos	3.1	Conceptos de programación orientada a objetos
		4.1	Lenguaje de pseudocódigo
		4.2	Aplicación del lenguaje de pseudocódigo
	El pseudocódigo en el diseño de programas informáticos.	4.3	Diseño de aplicaciones con pseudocódigo
		5.1	Análisis de sistemas informáticos
Elabora programas informáticos utilizando lenguajes de tercera y cuarta generación	Interpretación del pseudocódigo dentro del análisis de sistemas informáticos	5.2	Optimización de tamaño contra optimización de tiempo.
	Conociendo Visual Basic .Net 2010	6.1	Introducción
		6.2	Las Variables
		6.3	Los diferentes tipos de datos
		6.4	Las constantes
		6.5	Las Estructuras de control
		6.6	Los Operadores
	Programando Visual Basic .Net 2010	7.1	Instrucción de control for , do while loop , do loop while
		7.2	Operadores Relacionales Mayor que , Menor que , Mayor o igual que, Menor o igual que , No es igual que o es diferente que
		7.3	Instrucción IF
		7.4	Select Case
	Programando Controles Visual Basic .Net 2010	8.1	Listbox y dropdownlist , checkbox y checkboxlist , radiobutton y radiobuttonlist
		8.2	Depuración Código
		8.3	Trabajar con gráficos
		8.4	Manejo de mensajes y errores.
	Manejo de Archivos con Visual Basic .Net 2010	9.1	Introducción
		9.2	Los formatos (codificación) de los ficheros
		9.3	Leer y escribir ficheros de texto
		9.4	Anexar contenido a un fichero existente

8. Planificación secuencial del curso

Semana 1- 4					
# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
1	1.Conceptos Básicos de Programación	1.1 Los lenguajes de programación actuales 1.2 Algoritmo 1.3 Variables y constantes 1.4 Tipos de datos 1.5 Constante 1.6 Operadores 1.7 Funciones	(1) Socialización del Sílabo e Indicaciones Generales (1) Clases magistrales (2) Foro Virtual (2) Foro (3) Exposición del estudiantes (3) Exposición del estudiantes	(3) Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 1-20 Caro Pineda, Silvana	Portafolio De Prácticas de laboratorio Progreso 1: Fecha de entrega: Del 27 a 29 de marzo Portafolio de Ejercicios Progreso 1: Fecha de entrega: Del 27 a 29 de marzo Trabajos de apoyo Progreso 1: Fecha de entrega: Del 27 a 29 de marzo Portafolio de Ejercicios Progreso 1: Fecha de entrega: Del 27 a 29 de marzo
	2.Programación Estructurada	2.1 Conceptos de programación estructurada	(3) Exposición del estudiantes		Test de evaluación, teórico práctico Progreso 1: Fecha de entrega: Del 27 a 29 de marzo
	3.Programación Orientada a Objetos	3.1 Conceptos de programación orientada a objetos	(1) Clases magistrales		

Semana 5- 7					
# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
2	4. El pseudocódigo en el diseño de programas informáticos.	4.1 Lenguaje de pseudocódigo 4.2 Aplicación del lenguaje de pseudocódigo 4.3 Diseño de aplicaciones con pseudocódigo	(1) Clases magistrales (2) Talleres de ejercicios (2) Talleres de ejercicios	(3) Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 25-29 Caro Pineda, Silvana	Portafolio De Prácticas de laboratorio Progreso 1: Fecha de entrega: Del 17 al 19 de abril Portafolio de

	5.Interpretación del pseudocódigo dentro del análisis de sistemas informáticos	5.1Análisis de sistemas informáticos 5.2Optimización de tamaño contra optimización de tiempo.	(1)Talleres de ejercicios Presentación magistral (3)Proyectos (1)Prácticas de	(3)Proyecto de Apoyo de p. 33, 43 (Caro Pineda, Silvana)	Ejercicios Progreso 1: Fecha de entrega: Del 17 al 19 de abril Trabajos de apoyo Progreso 1: Fecha de entrega: Del 17 al 19 de abril Examen, teórico práctico Progreso 1: Fecha de entrega: Del 24 al 26 de abril
--	--	--	--	--	---

Semana 8- 9					
# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
3	6.Conociendo Visual Basic .Net 2010	6.1Introducción 6.2 Las Variables 6.3 Los diferentes tipos de datos 6.4 Las constantes 6.5 Las Estructuras de control	(1)Presentación magistral (3)Portafolio de Ejercicios (1)Prácticas de laboratorio. (1)Prácticas de laboratorio. (1)Prácticas de laboratorio.	(3)Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 57-58 (Foxall, James) (3)Trabajo de Apoyo de p. 350,351 (Foxall, James)	Portafolio De Prácticas de laboratorio Progreso 2: Fecha de entrega: Del 8 al 10 de mayo Portafolio de Ejercicios Progreso 2: Fecha de entrega: Del 8 al 10 de mayo Trabajos de apoyo Progreso 2: Fecha de entrega: Del 8 al 10 de mayo

		6.7 Los Operadores	(3)Portafolio de Ejercicios		Test teórico práctico Progreso 2: Fecha de entrega: Del 8 al 10 de mayo
--	--	--------------------	-----------------------------	--	--

Semana 10-11

# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
3	7. Programando Visual Basic .Net 2010	7.1 Instrucciones de control for , do while loop , do loop while	(1)Presentación magistral	(3)Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 415 (Foxall, James)	Portafolio De Prácticas de laboratorio Progreso 2: Fecha de entrega: Del 29 al 31 de mayo
		7.2 Operadores Relacionales Mayor que , Menor que , Mayor o igual que, Menor o igual que , No es igual que o es diferente que	(3)Portafolio de Ejercicios	(3)Proyecto de Apoyo de p. 377, (Foxall, James)	Portafolio de Ejercicios Progreso 2: Fecha de entrega: Del 29 al 31 de mayo
		7.3 Instrucción IF	(1)Prácticas de laboratorio.	(1)Trabajos de Apoyo de p. 398, (Foxall, James)	Trabajos de apoyo Progreso 2: Fecha de entrega: Del 29 al 31 de mayo
		7.4 Select Case	(1)Prácticas de laboratorio.	(1)Trabajos de Apoyo de p. 398, (Foxall, James)	Examen, teórico práctico Progreso 2: Fecha de entrega: Del 29 al 31 de mayo

Semana 12-14

# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
-------	------	----------	------------------------------	-------------------------	--------------------------------

3	8. Programando Controles Visual Basic .Net 2010	8.1 Listbox y dropdownlist , checkbox y checkboxlist , radiobutton y radiobuttonlist	(1)Presentación magistral	(3)Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 494 (Foxall, James)	Portafolio De Prácticas de laboratorio Evaluación Final: Fecha de entrega: Del 19 al 21 de junio
		8.2 Depurar Código ,	(2)Prácticas virtuales de laboratorio.	(2)Trabajos virtuales de Apoyo de p. 444 (Foxall, James)	Portafolio de Ejercicios Evaluación Final: Fecha de entrega: Del 19 al 21 de junio
		8.3 Trabajar con gráficos	(3)Portafolio de Ejercicios	(1)Trabajos de Apoyo de p. 518 (Foxall, James)	Trabajos de apoyo Evaluación Final: Fecha de entrega: Del 19 al 21 de junio
		8.4 Manejo de mensajes y errores.	(2)Prácticas de laboratorio virtuales.	(2)Trabajos virtuales de Apoyo de p. 494 (Foxall, James)	Examen, teórico práctico Evaluación Final: Fecha de entrega: Del 19 al 21 de junio

Semana 15-18

# RdA	Tema	Sub tema	Actividad/ metodología/clase	Tarea/ trabajo autónomo	MdE/Producto/ fecha de entrega
3	9. Manejo de Archivos con Visual Basic .Net 2010	9.1Introducción	(1)Presentación magistral	(3)Portafolio de Ejercicios , prácticas , lecturas y exposiciones p. 542 (Foxall, James)	Portafolio De Prácticas de laboratorio Evaluación Final Fecha de entrega: Del 10 al 19 de julio
		9.2Los formatos (codificación) de los ficheros Leer y escribir ficheros de	(2)Foro virtual	(1)Trabajos de Apoyo de p. 570 (Foxall, James)	Portafolio de Ejercicios Evaluación Final: Fecha de entrega: Del 10

		texto			al 19 de julio
		9.3 Anexar contenido a un fichero existente	(1)Prácticas de laboratorio.	(1)Trabajos de Apoyo de p. 670 (Foxall, James)	Examen, teórico práctico Evaluación Final: Fecha de entrega: Del 10 al 19 de julio Recuperaciones: Del 10 al 19 de julio

9. Normas y procedimientos para el aula

- ✓ El alumno podrá ingresar solo hasta 15 minutos después de la hora indicada, luego, podrá ingresara a la segunda hora.
- ✓ Se permitirá entregar una tarea hasta un día después de la fecha indicada con una penalidad del 50% de la nota asignada.
- ✓ Una vez detectado el plagio en ejercicios, exámenes, proyectos, y todas las actividades de aprendizaje solicitadas por el docente, se calificará con la mínima calificación (cero).
- ✓ Se restringe durante el desarrollo de clases el uso de celulares, redes sociales y audífonos, gorras, gafas, etc.
- ✓ No se recibirán trabajos fuera del aula virtual.
- ✓ No se podrá comer ni beber en al aula.
- ✓ Todas las instalaciones de la UDLA son declaradas, edificaciones libre de humo.
- ✓ En el caso de inasistencia es responsabilidad del estudiante igualarse en los contenidos de la materia dictada en dicha clase.
- ✓ En el caso de que un estudiante falte a una sesión en la que se realicen pruebas o prácticas de laboratorio, no se podrán recuperar las calificaciones bajo ningún concepto.

10. Referencias bibliográficas

10.1. Principales.

- Farrell, Joyce (2013). Introducción a la programación lógica y diseño. (7ma ed). México, D.F. Cengage Learning
- Domínguez Vera, Edgar Danilo (2014). Programación estructurada: raptor y lenguaje C. México Alfaomega

10.2. Referencias complementarias.

- López Román, Leobardo. (2011). Programación estructurada y orientada a objetos. México Alfaomega.

- Méndez Girón, Alejandra Margarita (2013). Diseño de algoritmos y su programación en C. México Alfaomega.

11. Perfil del docente

Ingeniero en Informática, Master en Educación mención en Educación Superior y Master en Ingeniería del Software por la Universidad de las Fuerzas Armadas (ESPE). Suficiencia en el idioma inglés por la Universidad de las Fuerzas Armadas (ESPE). Docente (Pregrado – Postgrado) de la Universidad de las Fuerzas Armadas (ESPE) y de la Universidad Tecnológica América. Operador de sistemas informáticos, Analista de Sistemas, Coordinador de Sistemas, Jefe de Sistemas, Director de Proyectos y Responsable del área tecnológica en la Veeduría Internacional para la reforma de la justicia Ecuatoriana, Rector del Instituto Tecnológico Superior Andrés F. Córdova de la ciudad de Quito y Rector encargado del Instituto Tecnológico Superior Consejo Provincial de Pichincha.