

## SILABO

### I: IDENTIFICACIÓN

**ESPACIO DE APRENDIZAJE:** PROGRAMACIÓN I  
**CÓDIGO:** CCT100

**UNIDAD DE COMPETENCIA:**

**DURACIÓN:** 60 Horas Totales  
HORAS TEORICAS 20 Horas Teóricas  
HORAS PRÁCTICAS 40 Horas Prácticas  
MODALIDAD: Unidades secuenciales

### II: DESCRIPCIÓN POR ÁREA DE FORMACIÓN Y PRERREQUISITO

Área de formación: Especialidad  
Ubicación en la malla: 1° Periodo  
Prerrequisito: MATT101 Introducción al Álgebra

**I: UNIDADES DE APRENDIZAJE**
**1º UNIDAD DIDACTICA:** FUNDAMENTOS DE LA PROGRAMACIÓN.

**DURACIÓN:** 6 Horas Teóricas 4, Horas Prácticas 2

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE:		Introducir al mundo de la programación de lenguajes de alto nivel, desarrollo de lógicas y expresiones aritméticas,								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 11 Semana 1	90 minutos	Identifica los términos de programación, algoritmos, lenguajes de programación de alto nivel	Comprende la diferencia entre programa y algoritmo.  Diferencia los lenguajes de programación de alto nivel.  Identifica elementos de un diagrama de flujo	Maneja los diferentes conceptos de programación, analiza casos sobre programas y algoritmos	Se interesa por utilizar los programas de alto nivel según sus aficiones	Resolución de casos  Trabajo colaborativa	Casos a resolver				(Villalobos, 2006)
Jueves 12 semana 1	90 minutos	Reconoce los operadores aritméticos, lógicos, y de asignación.	Identifica los operadores, y aplica su correcto uso, en expresiones lógicas, matemáticas y de asignación.	Ejecuta ejercicio de comprensión con los diferentes operadores	Muestra la diferencia entre cada operador	Desarrollar ejercicios de comprensión lógica	Ejercicios a resolver				

Técnico Universitario en Diseño y Desarrollo Web

Fecha octubre de 2017

Lunes 16 Semana 1	90 minutos	Elaborar un programa estructurado	Comprende la estructura general de un programa.  Distingue los tipos de Datos y Operadores	Analizar la estructura de un programa.  Resumir los tipos de datos y operadores.	Participa en la generación de una solución. Asigna los tipos de datos	Comparte soluciones, y utiliza las mejores prácticas estructuradas para resolver	Ejercicios a resolver				
Martes 17 Semana 1	90 minutos	Identificar funciones en requerimientos de sistema	Interpreta las funciones generadas por el usuario, los argumentos y parámetros formales, funciones con y sin argumentos y funciones que regresan valores.	Elaborar una función con o sin argumentos, que regrese valores.	Muestra el interés de encontrar la mejor práctica para la generación de funciones con o sin argumentos que regrese valores.	Resolución de problemas	Problemas a resolver				

**2º UNIDAD DIDACTICA: ESTRUCTURAS DE CONTROL 1**
**DURACIÓN: 6 Horas Teóricas 4, Horas Prácticas 2**

Fecha y Nº Semana	OBJETIVO DE APRENDIZAJE:		Generar lógica de programas que incluyan estructuras de decisión, interacción y control en un lenguaje de alto nivel								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 18 Semana 2	90 minutos	Comprende los que son las estructuras de control	Distingue las diferentes estructuras de control, secuenciales, Selectivas, Repetitivas y las diferentes alternativas	Analiza los problemas por medio de las diferentes estructuras y las alternativas simple o dobles	Comprende el uso de controles comunes	Trabajo colaborativo					(J, 2008)
Jueves 19 Semana 2	90 minutos	Comprende la estructura secuencial	Define la estructura secuencial	Analiza problemas por medio de esta estructura	Comprende el uso de controles secuencial	Trabajo colaborativo					
Lunes 23 Semana 2	90 minutos	Comprende la estructura selectiva	Define estructura de alternativa simple, alternativas dobles, y alternativas múltiples	Aplica la diferentes alternativas en las diferentes soluciones de los problemas propuestos	Comprende el uso de controles selectiva	Resolución de casos					
Martes 24 Semana 2	90 minutos	Comprende la estructura Repetitiva	Defina la estructura repetitiva	Aplica la estructura for como una solución a problemas de la comunes	Comprende el uso de estructura repetitiva	Resolución de casos					

**3ra UNIDAD DIDACTICA: ESTRUCTURAS DE CONTROL 2**
**DURACIÓN: 6 Horas Teóricas 4, Horas Prácticas 2**

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE: (según programa)		Generar lógica de programas que incluyan estructuras de decisión, interacción y control en un lenguaje de alto nivel								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 25 y Jueves 26 Semana 3	180 minutos	Conoce el concepto de arreglos y es capaz de crear arreglos unidimensionales	Conoce los arreglos unidimensionales, su concepto y forma general tanto para arreglos numéricos como los que se utilizan con caracteres	Analizar arreglos unidimensionales sean estos numéricos o de caracteres	Sensibiliza a la utilización de apuntadores dentro de arreglos sean estos	Resolución de casos					
Lunes 30 Semana 3	90 minutos	Conoce los conceptos de arreglos bidimensionales	Conoce los arreglos bidimensionales, su concepto y forma general tanto para arreglos numéricos como los que se utilizan con caracteres	Analizar arreglos bidimensionales sean estos numéricos o de caracteres	unidimensionales o bidimensionales						
Martes 31 Semana 3	90 minutos	Conoce el concepto de apuntadores y es capaz de crear apuntadores	Describe los apuntadores, los tipos, las operaciones con apuntadores, las relaciones con los arreglos.	Desarrollar apuntadores, aplicar para casos particulares los diferentes tipos de apuntadores,	Comprende la importancia de utilizar operadores para las mejores soluciones	Resolución de casos					

**4ta UNIDAD DIDACTICA: Seudocódigo**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE: (según programa)		Proponer y construir un programa integral en un lenguaje de alto nivel aplicado a una problemática específica de la ciencia o de la ingeniería.								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 1 Semana 4	90 minutos	Construir un programa integral en un lenguaje de alto nivel aplicado a una problemática específica de la ciencia o de la ingeniería.	Analiza el problema específico.	Elaborar un pseudo código que sea acorde al análisis realizado del problema específico presentado	Es consciente de la importancia del delicado trabajo de análisis, diseño, implementación y verificación	Resolución de casos					
Jueves 2 Semana 4	90 minutos		Interpreta del problema específico la codificación que podría diseñarse a través de un pseudo código	Traducir el pseudo código en la sintaxis del lenguaje de alto nivel que sea consistente y homogéneo.	Es consiente que utilizar estructuras de programación, ciclos, arreglos y apuntadores permite generar soluciones óptimas						
Lunes 6 Semana 4	90 minutos		Indica las pruebas modulares e integrales que pueden desarrollar con la codificación para lograr los resultados esperados	Ejecutar las pruebas modulares e integrales para determinar el grado de aplicabilidad, usabilidad e interactividad con el usuario	Entiende la importancia de Diseñar sistemas de acuerdo a las necesidades y requerimientos del problema específico						

**Técnico Universitario en Diseño y Desarrollo Web**

Fecha octubre de 2017

Martes 7 Semana 4	90 minutos		Relaciona las actividades de mantenimiento que requieren.	Crear un plan de pruebas y casos de pruebas del sistema	Entiende la importancia de crear pruebas para darle calidad al proyecto						
----------------------	------------	--	---	---	---	--	--	--	--	--	--

**5ta UNIDAD DIDACTICA: Lenguaje de Programación**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº Semana	OBJETIVO DE APRENDIZAJE: (según programa)		Generar programas que incluyan estructuras de decisión, interacción y control en un lenguaje de alto nivel								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 8 Semana 5	90 minutos	Comprender las características de un programa en un lenguaje de programación de alto nivel	Analiza la estructura básica de un programa en java	Elabora un código que le permite compilar su primer programa.	Aplica las buenas prácticas de codificación en lenguajes de programación de alto nivel	Ejercicios de clase					
Jueves 9 Semana 5	90 minutos	Traducción de pseudocódigo a lenguajes de programación estructura secuencial	Representa el pseudocódigo a los lenguajes de programación de una	Elabora programas en base a un análisis previo	Comprende la importancia de realizar análisis de sistema	Resolución de problemas					

Técnico Universitario en Diseño y Desarrollo Web

Fecha octubre de 2017

			estructura secuencial								
Lunes 13 Semana 5	90 Minutos	Traducción de pseudocódigo a lenguajes de programación estructura Selectiva	Representa el pseudocódigo a los lenguajes de programación de una estructura selectiva	Elabora programas en base a un análisis previo	Comprende la importancia de realizar análisis de sistema	Resolución de problemas					
Martes 14 Semana 5	90 Minutos	<b>EXAMEN I PARCIAL</b>									



**6ta UNIDAD DIDACTICA: Lenguaje de Programación**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE: (según programa)		Codificar las diferentes estructuras de control, y estructuras de datos en lenguajes de programación de alto nivel								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las <u>actividades</u> )	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 15 Semana 6	90 minutos	<b>Revisión de Examen</b>  Traducción de pseudocódigo a lenguajes de programación estructura Repetitiva	Representa el pseudocódigo a le lenguajes de programación de una estructura Repetitiva	Elabora programas en base a un análisis previo	Comprende la importancia de realizar análisis de sistema	Resolución de problemas					
Jueves 16 Semana 6	90 minutos	Traducción de pseudocódigo a lenguajes de programación arreglos	Representa el pseudocódigo a lenguajes de programación los arreglos	Elabora programas en base a un análisis previo	Comprende la implementación de arreglos en sus dimensiones						
Lunes 20 Semana 6	90 minutos										
Martes 21 Semana 6	90 minutos	Traducción de pseudocódigo a lenguajes de programación apuntadores	Representa el pseudocódigo a lenguajes de programación de apuntadores	Elabora programas en base a un análisis previo	Comprende la implementación de apuntadores						

**7ma UNIDAD DIDACTICA: Fundamentos a la programación orientada a Objetos**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	<b>OBJETIVO DE APRENDIZAJE:</b> (según programa)		Que el alumno pueda Diferenciar entre clases y objetos, Identificar atributos y métodos, Construir un diagrama de clase								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODO LÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 22 Semana 7	90 minutos	Conoce los fundamentos de la programación orientada a objetos	Comprende los conceptos claves de la programación orientada a Clase, objeto, atributos y métodos.	Analiza los conceptos de la POO	Aplica los conocimientos de la POO en desarrollo de aplicaciones	Mapas conceptuales					(Barnes, 2007)
Jueves 23 Semana 7	90 minutos	Diagrama clases	Entiende que es un diagrama de clase y los componentes básicos.	Diseña la representación gráfica, a través del diagrama de clase	Comprende la importancia de diagramas de clases en la POO	Estudio de casos					
Lunes 27 Semana 7	90 minutos	Conoce los pilares de la POO	Comprende los objetos, herencia, abstracción, polimorfismo, encapsulamiento	Analiza los programas en base a los pilares de POO	Aplica los conocimientos de la POO en desarrollo de aplicaciones	Cuadros comparativos					
Martes 28 Semana 7	90 minutos	Desarrollar conocimientos aprendidos de POO	Traducir la POO a lenguajes de programación	Analiza los programas en base a los pilares de POO	Aplica los conocimientos de la POO en desarrollo de aplicaciones	Resolución de problemas					

**8va UNIDAD DIDACTICA: Programación Orientada en Objetos**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	<b>OBJETIVO DE APRENDIZAJE:</b> (según programa)		El alumno implemento los conocimientos de programación orientada a objetos lenguajes de programación.								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGI CAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 29 Semana 8	90 minutos	Codificar clase, métodos, atributos.	Comprende la estructura de la clase, métodos, atributos en lenguajes de programación	Desarrolla clases, atributos, métodos	Aplica los conocimientos de la POO en desarrollo de aplicaciones	Resolución de problemas					
Jueves 30 Semana 8	90 minutos		Comprende lo que es la sobrecarga de métodos y la correcta aplicación	Desarrolla clases con sobrecarga de métodos							
Lunes 4 Semana 8	90 minutos		Entiende la Composición de objetos	Crea objetos simples y complejos							
Martes 5 Semana 8	90 minutos		Entiende lo que es herencia. Entiende la diferencia entre herencia y composición	Crea herencia en base a los objetos desarrollados							

**9na UNIDAD DIDACTICA: POO y Excepciones**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE: (según programa)		El alumno comprende la importancia de reutilización de objetos y de agregar excepciones en los métodos de cada clase.								
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGI CAS (Actividad a realizar)	RECURSOS: (Para las actividades )	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 6 Semana 9	90 minutos	Comprende la estructura de crear polimorfismo, y la importancia del mismo	Define en concepto de polimorfismo y la estructura en lenguaje de programación	Desarrolla objetos aplicando el polimorfismo	Se sensibiliza al comprender la importancia de crear extensibilidad al momento de programar, para que los proyectos sea duraderos	Resolución de problemas					
Jueves 7 Semana 9	90 minutos			Desarrolla ejercicios de clase para integrar conocimientos							
Lunes 13 Semana 9	90 minutos	Comprende lo que son las excepciones y su importancia	Reconoce los diferentes tipos de excepciones, y como crear excepciones	Desarrolla métodos aplicando las excepciones posibles.	Aplica excepciones en el desarrollo de sistemas.						
Martes 14 Semana 9	90 minutos										

**10ma UNIDAD DIDACTICA: Arquitectura de Software**
**DURACIÓN: 6 Horas Teóricas 2, Horas Prácticas 4**

Fecha y Nº  Semana	OBJETIVO DE APRENDIZAJE:										
	Nº de Horas de Clase	COMPETENCIA (Acción)	CONTENIDOS (Realidad concreta)			ESTRATEGIAS METODOLÓGICAS (Actividad a realizar)	RECURSOS: (Para las actividades)	INDICADOR DE LOGRO			Bibliografía
			Conceptual	Procedimental	Actitudinal			TIPO DE EVALUACION (Tipo/Instrumento)	VALOR ACUMULATIVO	FECHA	
Miércoles 13 Semana 10	90 minutos	Conoce lo que es lo que es arquitectura de software	Describe las arquitecturas de software más comunes	Plantea sistemas con arquitecturas de desarrollo de software de alto nivel	Aplica las mejores prácticas en base al desarrollo de arquitecturas de software	Exposiciones					
Jueves 14 Semana 9	90 minutos	Conoce a lo que es la arquitecturas orientada a servicios	Define lo que es SOA, sus ventajas y la nuevas tendencias de desarrollo	Crea arquitecturas basadas en servicios y micro servicios	Aplica diseño de arquitectura basa en servicios.	Exposiciones					
Lunes 18 Semana 10	90 minutos	<b>DEFENSA DE PROYECTO FINAL</b>									
Martes 19 Semana 10	90 minutos	<b>EXAMEN I PARCIAL</b>									

#### IV: ORIENTACIONES METODOLÓGICAS

- Iniciar el proceso de enseñanza-aprendizaje a partir de los conocimientos previos de los estudiantes. Diagnóstico.
- Centrar la docencia en el aprendizaje de los estudiantes, más que en la enseñanza. El estudiante debe ser activo.
- Situar y vincular permanentemente los aprendizajes, contenidos y actividades con el contexto social y laboral de los estudiantes y carrera que estudian.
- Utilizar la resolución de problemas como uno de los ejes fundamentales de la enseñanza-aprendizaje.
- Promover en los estudiantes la reflexión sobre sus conocimientos y las posibles implicaciones de sus actos.
- Promover aprendizajes de conocimientos, habilidades y actitudes, integrados y relevantes en el contexto de la carrera.

#### V: EVALUACIÓN DE UNIDADES

UNIDAD	TIPO DE EVALUACIÓN
1	Ejercicios de clase, trabajo colaborativos, resolución de casos
2	Ejercicios de clase, trabajo colaborativos, resolución de casos, exposiciones
Examen	Sumativa integradora
Deben existir, además, evaluaciones formativas (sin calificación) durante el proceso de aprendizaje.	

\*Evaluación parcial: corresponde a la evaluación de determinados aprendizajes de la unidad.

\*Evaluación sumativa integradora: corresponde a la evaluación que integra los aprendizajes relevantes de la unidad.

\*Evaluación formativa: corresponde a la evaluación de aprendizajes con el fin de retroalimentar el proceso de aprendizaje. No lleva calificación.

#### VI: BIBLIOGRAFÍA

Barnes, D. J. (2007). *Programación orientada a objetos Java*. Pearson.

J, P. (208). *Como programar a Java*. Deitel .

Villalobos, J. A. (2006). *Fundamentos de programación*. 1.



FIRMA DOCENTE:

\_\_\_\_Gabriela Lindalee Argueta Argueta\_\_\_\_\_