



Fundamentos de programación Modalidad Connect

Guía para el profesor
Clave LTTI1802
Nivel Profesional Ejecutivo

Contenido

Información general del curso.....	3
Competencias del curso	3
Introducción al curso	3
Metodología.....	3
Evaluación	4
Estructura de las sesiones.....	4
Actividades.....	5
Evaluación	7
Calendario	7
Contenido del curso.....	8
Herramientas.....	9
Preguntas frecuentes	9
Guía para las sesiones	10
Semana 1	10
Semana 2	14
Semana 3	19
Anexo 1. Rúbrica de evidencia 1	30
Anexo 2. Rúbrica evidencia 2	31

Información general del curso

Nombre del curso: Fundamentos de programación

Nivel: Profesional Ejecutivo

Plan académico: 2018

Modalidad: **Connect**

Clave: LTT11802

Competencias del curso

- Soluciona problemas utilizando un lenguaje de programación de última generación, aplicando la lógica y algoritmia en su solución.

Introducción al curso

¿Alguna vez has pensado cómo funcionan internamente los programas y aplicaciones que utilizas todos los días?

En la actualidad existe un gran número de lenguajes de programación que han evolucionado para que nuestra vida pueda estar acompañada de dispositivos móviles, Internet, redes sociales, videojuegos y todo aquello que necesita ser configurado internamente con el uso de programación.

A lo largo de este curso de Fundamentos de Programación aprenderás las bases de la programación, la lógica y algoritmia para la resolución de problemas a través de las mejores

prácticas en la industria del desarrollo de software, y con el uso de herramientas de vanguardia. Este será el inicio de tu trayectoria como futuro desarrollador de software.

¡Bienvenido!

Metodología

Se ha diseñado un curso **Connect** con la finalidad de ser impartido por un **docente líder con experiencia en el ámbito laboral**, quien compartirá su conocimiento, experiencia y las mejores prácticas que realiza en su labor profesional.

La experiencia de cursos Connect promueve la interacción virtual entre estudiantes localizados en diferentes campus de la Universidad Tecmilenio, como una forma de enriquecer su formación, contrastando la realidad de su ciudad o región con la de otros compañeros.

Durante cada sesión virtual, el docente transmite su experiencia y actúa como guía en el proceso de aprendizaje durante la realización de las actividades.

El curso es **tetramestral** y tiene una distribución **semanal**; en cada semana se lleva a cabo una sesión virtual sincrónica de tres horas a través de una herramienta tecnológica de videoconferencia. La asistencia del participante a estas sesiones de videoconferencia es muy importante, pero no obligatoria, ya que tiene la posibilidad de revisar la sesión grabada en caso de no poder asistir en el horario establecido.

Bibliografía

Cada curso requiere un material bibliográfico disponible para su compra. Para conocer cuál es el libro que el alumno debe adquirir, revisa la sección Bibliografía del curso.

Evaluación

En la sección Evaluación el alumno puede consultar cómo se integrará la calificación final del curso. Dependiendo del curso, la evaluación puede variar con una combinación de los siguientes elementos:

- Exámenes aplicados en plataforma en las semanas 1 y 3.
- Dos evidencias para acreditar el avance en el nivel de competencia adquirido por el alumno.
- Actividades que retomen el contenido conceptual de los temas de la semana.
- Evaluación final estandarizada compuesta por instrumentos tales como mini casos, exámenes de opción múltiple, ensayos, proyectos, entre otros.

Estructura de las sesiones

Las sesiones se dividen en tres bloques; estas son las actividades que se recomienda realizar:

Bloque 1	Bloque 2	Bloque 3
<ul style="list-style-type: none"> ● Bienvenida y presentación de agenda. ● Actividad de bienestar-mindfulness. ● Desarrollo de temas de la semana. <ul style="list-style-type: none"> ○ Aplicación en contextos reales. ○ Actividades. ○ Cierre del tema. ● Cierre del bloque mediante utilización de herramientas tecnológicas diversas para una dinámica, foro o quiz. 	<ul style="list-style-type: none"> ● Actividad de reconexión. ● Desarrollo de temas de la semana. <ul style="list-style-type: none"> ○ Aplicación en contextos reales. ○ Actividades. ○ Cierre del tema. ● Cierre del bloque mediante utilización de herramientas tecnológicas diversas para una dinámica, foro o quiz. ● Receso. 	<ul style="list-style-type: none"> ● Actividad de reconexión. ● Desarrollo de temas de la semana. <ul style="list-style-type: none"> ○ Aplicación en contextos reales. ○ Actividades. ○ Cierre del tema. ● Cierre de la sesión mediante utilización de herramientas tecnológicas diversas para una dinámica, foro o quiz.

- Receso.

Antes de acudir a una sesión, es necesario que el alumno realice las lecturas de las explicaciones y del libro de texto, ya que le proporcionarán los fundamentos teóricos de los temas del curso. De igual manera, se requiere que revise el material adicional como videos y lecturas.

Durante las sesiones sincrónicas el docente da una breve explicación del tema, resuelve dudas, comparte las instrucciones de las actividades y te acompaña durante la realización de estas.

Actividades

Algunas actividades han sido diseñadas para realizarse de manera individual y otras de manera colaborativa. Para las actividades colaborativas, tú como profesor deberás integrar equipos con alumnos de diferentes campus, lo cual te permite obtener experiencias de aprendizaje más enriquecedoras.

Para mayor efectividad del trabajo colaborativo se utilizan las funcionalidades de la herramienta de colaboración que permiten la creación de salas virtuales interactivas, donde puedes compartir pantallas, documentos, videos y audios.

Como una forma de promover el dinamismo y la interacción de los alumnos en distintos formatos, durante las sesiones puedes alternar intervenciones individuales, plenarias y grupales que enriquecen sus puntos de vista y al mismo tiempo les dan la oportunidad de presentar sus ideas y posturas en torno a los temas de clase.

El resultado de todas las actividades y tareas realizadas durante la semana deberá concentrarse en un solo documento, el cual el alumno lo entregará a través de la plataforma tecnológica para su revisión y evaluación por parte del docente.

Es muy importante que el alumno revise el esquema de evaluación y los criterios que utilizarás para otorgarle una calificación. Lo anterior con la intención de que desde el inicio de la semana tenga claro el nivel de complejidad y esfuerzo que se requiere para realizar las entregas semanales y garantizar el éxito dentro del curso.

En caso de tener dudas sobre algún ejercicio o sobre el contenido del curso, el alumno puede contactarte a través de los medios que le indiques.

Sesiones virtuales

Para la transmisión de las sesiones se utiliza una herramienta de videoconferencias. Con el fin de mejorar la calidad de dichas interacciones, se recomienda lo siguiente.

Es muy importante que cuentes con los siguientes **requerimientos tecnológicos** para llevar a cabo y con éxito las sesiones:



Red con conexión ancha para tener una excelente comunicación, mínimo con **6 MB** de ancho de banda.



Uso de los navegadores Chrome o Firefox



Computadora



Durante la sesión se recomienda no tener otras aplicaciones abiertas (ejemplo: Facebook, Netflix, YouTube, etc.)



Cámara



Micrófono o bocinas



Evaluación

Número	Evaluable	Ponderación
1	Evaluable semana 1	10
2	Evaluable semana 2	35
3	Evaluable semana 3	15
4	Evaluable semana 4	35
Total:		100

Calendario ✓

Semana	Temas	Actividades	Tareas	Evidencia	Examen
1	<ul style="list-style-type: none"> Tema 1. Conceptos básicos de programación y algoritmos Tema 2. Proceso de desarrollo y herramientas de programación Tema 3. Diagramas de flujo y pseudocódigos Tema 4. Estructura básica de un programa 	✓			
2	<ul style="list-style-type: none"> Tema 5. Aplicación del proceso de desarrollo en la solución de un problema Tema 6. Variables y asignaciones Tema 7. Entradas y salidas de datos 			✓	
3	<ul style="list-style-type: none"> Tema 8. Operadores y control de flujo Tema 9. Estatutos condicionales Tema 10. Estatutos de repetición Tema 11. Funciones predefinidas de un lenguaje 	✓			
4	<ul style="list-style-type: none"> Tema 12. Funciones y parámetros Tema 13. Arreglos unidimensionales Tema 14. Arreglos multidimensionales Tema 15. Manipulación de archivos 			✓	

Bibliografía y recursos especiales

Libros de texto

- Joyanes, J. (2013). Fundamentos Generales de Programación. México: McGraw-Hill. ISBN: 978-607-15-0818-8
- Ceruzzi, P. (2018). Breve Historia de la Computación. México: Fondo de Cultura Económica.

Libros de apoyo

- Guijarro, A., García, P., y Yanza, A. (2018). Organización y Arquitectura Computacional. Un enfoque práctico. Cuba: Editorial Académica Universitaria y Opuntia Brava.

Requerimientos

Software

- Raptor Flowchart Interpreter
- Visual Studio 2019

Contenido del curso

Tema 1.	Conceptos básicos de programación y algoritmos
Tema 2.	Proceso de desarrollo y herramientas de programación
Tema 3.	Diagramas de flujo y pseudocódigos
Tema 4.	Estructura básica de un programa
Tema 5.	Aplicación del proceso de desarrollo en la solución de un problema
Tema 6.	Variables y asignaciones
Tema 7.	Entradas y salidas de datos
Tema 8.	Operadores y control de flujo
Tema 9.	Estatutos condicionales
Tema 10.	Estatutos de repetición
Tema 11.	Funciones predefinidas de un lenguaje
Tema 12.	Funciones y parámetros
Tema 13.	Arreglos unidimensionales
Tema 14.	Arreglos multidimensionales
Tema 15.	Manipulación de archivos

Herramientas

Para asegurar que el alumno aproveche al máximo su experiencia educativa en esta modalidad de cursos, recomendamos que revise estos [tutoriales](#).

Preguntas frecuentes

¿En dónde o a quién reporto un error detectado en el contenido del curso?

Lo puedes reportar a través del botón **Mejora tu curso**, también puedes compartir sugerencias para el contenido y actividades del curso.

¿Quién me informa de la cantidad de sesiones y tiempo de cada sesión en las semanas?

El coordinador docente te debe de proporcionar esta información.

¿En qué semanas se aplican los exámenes parciales y el examen final?

Consulta con tu coordinador docente los calendarios de acuerdo con la modalidad de impartición.

¿Tengo que capturar las calificaciones en banner y en la plataforma educativa?

Sí, es importante que captures calificaciones en la plataforma para que los alumnos estén informados de su avance y reciban retroalimentación de parte tuya de todo lo que realizan en el curso. En banner es el registro oficial de las calificaciones de los

Guía para las sesiones

Semana 1

Bloque 1

Actividad	Descripción	Duración
Bienvenida y presentación de la agenda	El profesor se presenta ante el grupo y da una breve introducción al curso. El profesor explicará a los alumnos los contenidos y actividades que se revisarán durante la clase.	10 minutos
Actividad de bienestar	El profesor seguirá las instrucciones de la actividad correspondiente y accederá al siguiente enlace https://youtu.be/C2J0ldXHrwg	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 1. Conceptos básicos de programación y algoritmos Tema 2. Proceso de desarrollo y herramientas de programación	20 minutos
Actividad del tema	Realizar parte 1 de la Actividad de la semana.	15 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	10 minutos



Actividad 1

Parte 1

1. Realiza el diagrama de flujo en Raptor para el siguiente pseudocódigo:

Problema:

Dada la velocidad de un automóvil expresada en kilómetros por hora, desplegar la velocidad expresada en metros por segundo.

Pseudocódigo:

CONVERTIR VELOCIDAD A METROS POR SEGUNDO

Inicio

▷ Pedir al usuario la velocidad en kilómetros por hora

velocidad ← lectura del teclado

resultado = (velocidad * 1000) / 3600

▷ Imprimir resultado

Fin

Entregables

Reporte que incluya la actividad desarrollada.

Bloque 2

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad de preferencia física para ayudar al alumno a reconectarse a la clase (ejemplo, sentadillas, estiramientos, etcétera).	5 minutos
Desarrollo de temas de la semana	Revisión de los temas Tema 3. Diagramas de flujo y pseudocódigos	20 minutos
Actividad del tema	Realizar parte 2 de la Actividad de la semana.	20 minutos
Cierre de bloque	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	5 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	10 minutos



Actividad 1

Parte 2

2. Escribe el pseudocódigo que le corresponde dado el siguiente algoritmo:

Problema:

Dados dos valores, determinar si uno de ellos es múltiplo del otro.

Algoritmo:

- Leer el primer número (numero1).
- Leer el segundo número (numero2).
- Obtener el residuo de dividir numero1 entre numero2 ($\text{residuo1} = \text{numero1} \% \text{numero2}$).
- Obtener el residuo de dividir numero2 entre numero1 ($\text{residuo2} = \text{numero2} \% \text{numero1}$).
- Si el residuo1 es igual a cero, entonces el numero1 es múltiplo del numero2, por lo que se debe desplegar el mensaje “el primer número es múltiplo del segundo”.
- Si el residuo2 es igual a cero, entonces el numero2 es múltiplo del numero1, por lo que se debe desplegar el mensaje “el segundo número es múltiplo del primero”.

Entregables

Reporte que incluya la actividad desarrollada.

● ● ● ● **Bloque 3** ● ● ● ●

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad que refuerce alguna fortaleza. Intervenciones positivas.	5 minutos
Desarrollo de temas de la semana	Revisión de los temas Tema 4. Estructura básica de un programa	20 minutos
Actividad del tema	Realizar parte 3 de la Actividad de la semana.	30 minutos
Cierre de la sesión	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	5 minutos



Actividad 1

Parte 3

3. Genera el pseudocódigo y el diagrama de flujo para cada uno de los siguientes problemas. Para construir el diagrama utiliza Raptor, además de la imagen del diagrama de flujo debes incluir una imagen de la consola en la que se aprecie que el diagrama resuelve el problema en cuestión.

- Determinar la hipotenusa de un triángulo rectángulo, dado que el usuario proporciona de entrada las longitudes de sus catetos.
- Dado un número, indicar si es par-positivo, par-negativo, impar-positivo o impar-negativo.
- Dada una lista de valores enteros, determinar cuál es el mayor. Los valores son datos de entrada al programa que serán tecleados por el usuario uno a uno. El último será un cero.

4. Resuelve los siguientes problemas:

- Determina el área y volumen de un cilindro, dado su radio (R) y altura (H).
- Dados tres números, determina cuál es el más pequeño.
- Dada la calificación final de los alumnos de un grupo, despliega la cantidad de aprobados, la cantidad de reprobados y el promedio del grupo. La cantidad de alumnos será también un dato de entrada al programa

Entregable

Reporte que incluya la actividad desarrollada.

El resultado de todas las actividades y tareas realizadas durante la semana se deberá concentrar en **un solo documento** que se entregará a través de la plataforma tecnológica para su revisión y evaluación por parte del docente, con base en los siguientes criterios de evaluación.

Criterios de evaluación de la semana

Criterios de evaluación	Ponderación	Puntos sobre evaluación final
Criterio 1 <ul style="list-style-type: none"> • Construcción correcta del diagrama de flujo que resuelve el problema solicitado. 	30	4
Criterio 2 <ul style="list-style-type: none"> • Construcción correcta del pseudocódigo que resuelve el problema solicitado. 	30	4
Criterio 3 <ul style="list-style-type: none"> • Generación del pseudocódigo y diagrama de flujo para cada uno de los tres problemas solicitados. 	20	4

Criterio 4	<ul style="list-style-type: none"> Resolución de los problemas 	20	3
	Totales:	100	15

Semana 2

Bloque 1

Actividad	Descripción	Duración
Bienvenida y presentación de la agenda	El profesor se presenta ante el grupo y da una breve introducción a la sesión. El profesor explicará a los alumnos los contenidos y actividades que se revisarán durante la clase.	10 minutos
Actividad de bienestar	El profesor seguirá las instrucciones de la actividad correspondiente y accederá al siguiente enlace https://youtu.be/hSq87Fz0LY	5 minutos
Desarrollo de temas de la semana	Revisión de los temas Tema 5. Aplicación del proceso de desarrollo en la solución de un problema	20 minutos
Actividad del tema	Realizar la parte 1 de la Evidencia de la semana.	10 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	15 minutos



Evidencia 1

Parte 1

1. Lee con atención plena el siguiente escenario:

Trabajas para una empresa de desarrollo de software, y te piden dar una solución que se encargue de administrar cajeros automáticos.

Los casos de uso son los siguientes:

GUÍA PARA EL PROFESOR

Derechos reservados © Universidad Tecmilenio

- Retiro de dinero: el cliente podrá retirar su dinero siempre y cuando cuente con los fondos suficientes, el número de tarjeta y PIN coincidan, y su tarjeta no esté vencida.
- Transferencia de fondos: transferir dinero entre diferentes cuentas que se encuentren registradas como válidas. Se podrá realizar la transferencia de fondos, siempre y cuando la cantidad solicitada se tenga en la cuenta.
- Depósito de fondos: para depositar dinero a una cuenta se requiere el número de tarjeta a depositar y el monto. Se aceptan billetes únicamente, se debe especificar el monto a depositar, solicitar introducir billetes y realizar la validación, en caso de incumplir, se debe permitir modificar el monto a depositar.
- Consulta de saldos: el usuario podrá consultar en pantalla su saldo, solo si su número de tarjeta coincide con su PIN. En caso de que la tarjeta esté vencida, se debe mostrar en pantalla el siguiente texto: "Su plástico ha vencido, pase a ventanilla para obtener uno nuevo".

El banco nos ha proporcionado la siguiente información que representa los datos que ellos manejan:

Titular	Número	CVC	Fecha	Saldo
Juan Montes	4242424242424240	3 dígitos	Cualquier fecha	Cualquier saldo de prueba
Pedro Zapata	4000056655665550	3 dígitos	Cualquier fecha	Cualquier saldo de prueba
Ana Martínez	5555555555554440	3 dígitos	Cualquier fecha	Cualquier saldo de prueba
Rogelio Guerra	2223003122003220	3 dígitos	Cualquier fecha	Cualquier saldo de prueba

2. Con base en tus conocimientos, indica los pasos del ciclo de vida del desarrollo de un software.

3. Desarrolla el pseudocódigo y diagrama de flujo que responda a ofrecer una solución. La solución deberá contener lo siguiente:

- Análisis: examina los requerimientos y genera una lista de todas las necesidades detalladas que identifiques del sistema. Define las entradas, salidas y el proceso.
- Diseño: plantea el algoritmo de la solución utilizando pseudocódigo y diagramación en Raptor. Valida el diagrama de flujo y muestra las evidencias de la consola de Raptor. Indica también los casos de prueba para asegurar que tu programa funciona correctamente (mínimo tres casos de prueba, cada prueba deberá tener un identificador, la descripción de la prueba, las entradas y las salidas esperadas).

- Pruebas: ejecuta las pruebas y documenta la evidencia de cada caso de prueba. Genera el reporte de la ejecución de pruebas realizadas en Raptor.

Recuerda que el programa deberá usar las mejores prácticas de indentación y orden del código.

4. Con base en lo anterior, elabora un reporte del procedimiento que realizaste para desarrollar el software y los problemas que presentaste para lograr su correcto funcionamiento.

Nota: No olvides incluir en el documento tu algoritmo, pseudocódigo, capturas de pantalla de cada prueba, así como el diagrama de flujo en Raptor y los archivos fuente (el .rap para el diagrama de flujo y el .psc si utilizaste PSeInt).

Entregables

Reporte que incluya el desarrollo de la actividad planteada.

Bloque 2

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad de preferencia física para ayudar al alumno a reconectarse a la clase (ejemplo, sentadillas, estiramientos, etcétera).	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 6. Variables y asignaciones	20 minutos
Actividad del tema	Realizar parte 2 de la Evidencia de la semana.	15 minutos
Cierre de bloque	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	5 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	15 minutos



Evidencia 1

Parte 2

5. Una vez que tengas la primera fase y hayas desarrollado el algoritmo y diagrama de flujo, es tiempo de codificar utilizando las estructuras condicionales y de repetición que aprendiste. La solución deberá contener lo siguiente:

- Codificación: programa la solución utilizando C#, incluyendo comentarios de encabezado donde muestres lo que realiza el programa, las entradas y las salidas (puedes incluir, de preferencia, comentarios simples dentro del programa para mejorar el entendimiento de tu programa).
- Pruebas: ejecuta las pruebas y documenta la evidencia de cada caso de prueba. Genera el reporte de la ejecución de pruebas.

Recuerda que el programa deberá usar las mejores prácticas de indentación y orden del código.

6. Elabora un reporte con el procedimiento que realizaste para desarrollar el software y los problemas que presentaste para lograr su correcto funcionamiento.

Nota: Recuerda incluir en tu reporte lo que hiciste en la primera evidencia (pseudocódigo y diagrama de flujo), así como el código fuente (archivo .cs) y capturas de pantalla de cada una de tus pruebas de escritorio.

Entregables

Reporte que incluya el desarrollo de la actividad planteada.

Bloque 3

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad que refuerce alguna fortaleza. Intervenciones positivas.	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 7. Entradas y salidas de datos	25 minutos

Actividad del tema	Realizar parte 3 de la Evidencia de la semana.	20 minutos
Cierre de la sesión	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	10 minutos



Evidencia 1

Parte 3

7. Lee con atención plena el siguiente escenario:

La gente del banco está muy feliz con tus entregables, por motivos de seguridad piden manejar ellos mismos las entradas de información, es decir, el banco gestionará la información respecto a los datos de los tarjetahabientes y la tendrá disponible en un archivo de texto que el sistema deberá llamar cada vez que se realice una operación.

También piden que cuando el cliente verifique su saldo pueda imprimir su comprobante en un archivo de texto, mismo que el proveedor de infraestructura tendrá configurado para imprimirlo físicamente (nosotros interpretamos que enviarlo a un archivo de texto es lo mismo que imprimir).

Al conocer cómo programar con arrays, funciones, y sabiendo manejar archivos, podrás desarrollar un software mucho más robusto. Teniendo en cuenta los dos entregables pasados, en esta entrega final deberás realizar lo siguiente:

- Tomarás lo que ya desarrollaste y realizarás las adaptaciones necesarias utilizando dentro de la solución: arreglos, funciones, manipulación de archivos, ciclos repetitivos, estructuras condicionales, adecuado manejo de variables y constantes, así como cálculos para poder realizar cada operación que el cliente tendrá disponible en el sistema de cajero automático. La solución deberá contener lo siguiente:
- Análisis: examina los requerimientos que han sufrido alguna modificación. Define las entradas, procesos y salidas tomando en cuenta que los datos serán leídos de un archivo de texto, y que en la consulta de saldos también se hará uso de un archivo de texto que posteriormente será impreso.
- Diseño: revisa el algoritmo de solución general del programa y los algoritmos para la lógica de cada uno de los aspectos del proyecto. Deberás tomar en cuenta el diseño modular para dividir la solución en subpartes a través del uso de funciones. Genera también el diseño de casos de prueba (mínimo cinco) que te permitan probar todos los escenarios que pide el programa; no olvides tomar en cuenta los casos de prueba en caso de que algún archivo no

exista, o que el formato de entrada sea el incorrecto (cada prueba deberá contener un identificador, la descripción de la prueba, las entradas y las salidas esperadas).

- Codificación: programa la solución usando C#, incluye el comentario de encabezado de cada una de las funciones, en donde se muestre el objetivo de la función, la definición de parámetros y los retornos (si aplican). El programa deberá realizarse aplicando arreglos y archivos.
- Pruebas: ejecuta las pruebas y documenta la evidencia de cada caso de prueba. Genera el reporte de la ejecución de pruebas.

Recuerda que el programa debe usar las mejores prácticas de documentación e indentación de código.

8. Elabora un reporte con el procedimiento que seguiste para desarrollar el software y los problemas que presentaste para lograr el correcto funcionamiento.

Entregables

Documento que incluya el desarrollo de la actividad planteada.

El resultado de todas las actividades y tareas realizadas durante la semana se deberá concentrar en **un solo documento** que se entregará a través de la plataforma tecnológica para su revisión y evaluación por parte del docente, con base en la rúbrica de evaluación disponible en el Anexo 1.

● ● ● ● **Criterios de evaluación de la semana** ● ● ● ●

Ver Anexo 1

Semana 3

● ● ● ● **Bloque 1** ● ● ● ●

Actividad	Descripción	Duración
Bienvenida y presentación de la agenda	El profesor se presenta ante el grupo y da una breve introducción a la sesión. El profesor explicará a los alumnos los contenidos y actividades que se revisarán durante la clase.	10 minutos
Actividad de bienestar	El profesor seguirá las instrucciones de la actividad correspondiente y accederá al siguiente enlace https://youtu.be/IU6cDVjjqgc	5 minutos
Desarrollo de temas de la semana	Revisión de los temas Tema 8. Operadores y control de flujo Tema 9. Estructuras condicionales	20 minutos

Cierre de bloque	Realizar la parte 1 de la Actividad 2.	10 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	15 minutos



Actividad 2

Parte 1

Realiza los siguientes ejercicios relacionados con las estructuras de repetición for, while y do...while.

1. Indica qué despliega cada una de las siguientes secuencias de programación y después comprueba tu respuesta ejecutando el código.

<pre>for (int k = 5; k >= 5; k--) { Console.WriteLine("A"); }</pre>	Despliega:
<pre>for (int k = 5; k >= 5; k--) { Console.WriteLine("A"); Console.WriteLine(k); }</pre>	Despliega:
<pre>int i = 10; while (i < 13) { Console.WriteLine("A"); } Console.WriteLine("BB");</pre>	Despliega:
<pre>int i = 2; do { Console.WriteLine("A-"); i++; } while (i < 8);</pre>	Despliega:
<pre>int m = 20; do { m = m / 2;</pre>	Despliega:

```
Console.Write(m + " ");
} while (m >= 0);
```

```
int m = 3;
while (m < 21)
{
Console.Write("A" + m);
m += 3;
}
```

Despliega:

2. Escribe en los rectángulos el valor o expresión que se necesita para que la sección del programa realice la actividad especificada en la primera columna.

Lo que debe hacer la sección del programa	Sección del programa
Desplegar en pantalla el resultado de obtener la siguiente suma: 7+6+5+4+3	<pre>int suma = 0; for(int i = ____; ____; i--) { suma += i; } Console.Write(suma);</pre>
Al final del ciclo, la variable num debe contener un valor entre 100 y 999 incluyéndolos a ambos.	<pre>int num; do { Console.Write("A" + m); num = Int32.Parse(Console.ReadLine()); } while (____ ____ ____);</pre>
Desplegar en la pantalla la siguiente secuencia de valores: (1, 10) (2, 9) (3, 8) (4, 7) (5, 6)	<pre>for(int a = ____, b=10; a < b; ____, ____) { Console.WriteLine("(" + a + ", " + b + ")"); }</pre>

Entregable

Reporte que incluya el análisis del caso planteado.

• • • • **Bloque 2** • • • •

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad de preferencia física para ayudar al alumno a reconectarse a la clase (ejemplo, sentadillas, estiramientos, etcétera).	5 minutos
Desarrollo de temas de la semana	Revisión de los temas Tema 10. Estatutos de repetición	20 minutos
Actividad del tema	Realizar parte 2 de la Actividad de la semana	15 minutos
Cierre de bloque	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	5 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio	15 minutos



Actividad 2

Parte 2

3. Escribe las secciones del programa que se solicitan.

Lo que debe hacer la sección del programa	Sección del programa
a. Desplegar en pantalla los valores pares en el rango 2 de 100.	
b. Obtener la suma de todos los valores que teclee el usuario. Se sabe que el último número que tecleará el usuario será un cero.	

- c. Contar cuántos valores dados por el usuario fueron necesarios para que su suma genere un valor mayor a 5,000.

4. Realiza el análisis, diseño e implementación de un programa que lea un número entero N, que representa la cantidad de números enteros a pedir, y para cada número indica si el número es par o impar. Posteriormente, debe de indicar cuántos números fueron pares y cuántos fueron impares.

Ejemplo de la ejecución del programa:

```

¿Cuántos números serán? 5
Valor del número 1: 5
El número 5 es impar
Valor del número 2: 12
El número 12 es par
Valor del número 3: 34
El número 34 es par
Valor del número 4: 7
El número 7 es impar
Valor del número 5: 1
El número 1 es impar

La cantidad de números pares es 2
La cantidad de números impares es 3
  
```

Entregable

Reporte con el desarrollo de la investigación.

Bloque 3

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad que refuerce alguna fortaleza. Intervenciones positivas.	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 11. Funciones predefinidas de un lenguaje	30 minutos
Actividad del tema	Realizar la parte 3 de la Actividad de la semana.	15 minutos

Cierre de la sesión

El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.

10 minutos



Actividad 2

Parte 3

5. Realiza el análisis, diseño e implementación de un programa que pregunte el tipo de boleto (A, N, I), muestre su precio y luego pregunte si hay otro cliente, si el usuario contesta que sí, debe repetir el proceso. Considera lo siguiente:

Un parque de diversiones tiene una máquina para vender boletos de tres tipos:

- A Adulto que cuestan \$120
- N Niños que cuestan \$75
- I Adultos con credencial del INSEN que cuestan \$60

Para comprar los boletos en esta máquina debe solicitarse uno a la vez. Puedes suponer que la máquina tiene una cantidad ilimitada de boletos.

Ejemplo de la ejecución del programa:

```
Tipo de boleto: A
El boleto cuesta: $120.00
Otro boleto (s/n)? s
Tipo de boleto: I
El boleto cuesta: $60.00
Otro boleto (s/n): n
Fin del programa
```

6. Realiza el análisis, diseño e implementación de un programa que lea del teclado números enteros y los vaya sumando. Debe detenerse cuando la suma de los números leídos sea 1,000 o más, y mostrar el total de la suma y la cantidad de números que se sumaron.

Entregable

Reporte que integre el análisis de las situaciones planteadas.

GUÍA PARA EL PROFESOR

El resultado de todas las actividades y tareas realizadas durante la semana se deberá concentrar en **un solo documento** que se entregará a través de la plataforma tecnológica para su revisión y evaluación por parte del docente, con base en los siguientes criterios de evaluación.

Criterios de evaluación de la semana

	Criterios de evaluación	Ponderación	Puntos sobre evaluación final
Criterio 1	<ul style="list-style-type: none"> Desarrolla programas que emplean la estructura de repetición for para resolver problemas. 	20	5
Criterio 2	<ul style="list-style-type: none"> Desarrolla programas que emplean la estructura de repetición while para resolver problemas. 	30	5
Criterio 3	<ul style="list-style-type: none"> Desarrolla programas que emplean la estructura de repetición do...while para resolver problemas. 	40	5
Totales:		100	15

Semana 4

Bloque 1

Actividad	Descripción	Duración
Bienvenida y presentación de la agenda	El profesor se presenta ante el grupo y da una breve introducción a la sesión. El profesor explicará a los alumnos los contenidos y actividades que se revisarán durante la clase.	10 minutos
Actividad de bienestar	El profesor seguirá las instrucciones de la actividad correspondiente y accederá al siguiente enlace https://youtu.be/r-ctqMZnCd8	5 minutos

Desarrollo de temas de la semana	Revisión del tema Tema 12. Funciones y parámetros Tema 13. Arreglos unidimensionales	20 minutos
Cierre de bloque	Realizar parte 1 de la Evidencia 2.	10 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	15 minutos



Evidencia 2

Parte 1

1. Lee con atención plena el siguiente caso:

Trabajas para una empresa de desarrollo de software, y te piden dar una solución que se encargue de administrar cajeros automáticos.

Los casos de uso son los siguientes:

- Retiro de dinero: el cliente podrá retirar su dinero siempre y cuando cuente con los fondos suficientes, el número de tarjeta y PIN coincidan, y su tarjeta no esté vencida.
- Transferencia de fondos: transferir dinero entre diferentes cuentas que se encuentren registradas como válidas. Se podrá realizar la transferencia de fondos, siempre y cuando la cantidad solicitada se tenga en la cuenta.
- Depósito de fondos: para depositar dinero a una cuenta se requiere el número de tarjeta a depositar y el monto. Se aceptan billetes únicamente, se debe especificar el monto a depositar, solicitar introducir billetes y realizar la validación, en caso de incumplir, se debe permitir modificar el monto a depositar.
- Consulta de saldos: el usuario podrá consultar en pantalla su saldo, solo si su número de tarjeta coincide con su PIN. En caso de que la tarjeta esté vencida, se debe mostrar en pantalla el siguiente texto: "Su plástico ha vencido, pase a ventanilla para obtener uno nuevo".
- El banco nos ha proporcionado la siguiente información que representa los datos que ellos manejan:

Titular	Número	CVC	Fecha	Saldo
Juan Montes	4242424242424240	3 dígitos	Cualquier fecha	Cualquier saldo de prueba
Pedro Zapata	4000056655665550	3 dígitos	Cualquier fecha	Cualquier saldo de prueba

Ana Martínez	5555555555554440	3 dígitos	Cualquier fecha	Cualquier saldo de prueba
Rogelio Guerra	2223003122003220	3 dígitos	Cualquier fecha	Cualquier saldo de prueba

- La gente del banco está muy feliz con tus entregables, por motivos de seguridad piden manejar ellos mismos las entradas de información, es decir, el banco gestionará la información respecto a los datos de los tarjetahabientes y la tendrá disponible en un archivo de texto que el sistema deberá llamar cada vez que se realice una operación.

También piden que cuando el cliente verifique su saldo pueda imprimir su comprobante en un archivo de texto, mismo que el proveedor de infraestructura tendrá configurado para imprimirlo físicamente (nosotros interpretamos que enviarlo a un archivo de texto es lo mismo que imprimir).

Al conocer cómo programar con arrays, funciones, y sabiendo manejar archivos, podrás desarrollar un software mucho más robusto. Teniendo en cuenta los dos entregables pasados, en esta entrega final deberás realizar lo siguiente:

1. Tomarás lo que ya desarrollaste y realizarás las adaptaciones necesarias utilizando dentro de la solución: arreglos, funciones, manipulación de archivos, ciclos repetitivos, estructuras condicionales, adecuado manejo de variables y constantes, así como cálculos para poder realizar cada operación que el cliente tendrá disponible en el sistema de cajero automático. La solución deberá contener lo siguiente:
 - Análisis: examina los requerimientos que han sufrido alguna modificación. Define las entradas, procesos y salidas tomando en cuenta que los datos serán leídos de un archivo de texto, y que en la consulta de saldos también se hará uso de un archivo de texto que posteriormente será impreso.
 - Diseño: revisa el algoritmo de solución general del programa y los algoritmos para la lógica de cada uno de los aspectos del proyecto. Deberás tomar en cuenta el diseño modular para dividir la solución en subpartes a través del uso de funciones. Genera también el diseño de casos de prueba (mínimo cinco) que te permitan probar todos los escenarios que pide el programa; no olvides tomar en cuenta los casos de prueba en caso de que algún archivo no exista, o que el formato de entrada sea el incorrecto (cada prueba deberá contener un identificador, la descripción de la prueba, las entradas y las salidas esperadas).
 - Codificación: programa la solución usando C#, incluye el comentario de encabezado de cada una de las funciones, en donde se muestre el objetivo de la función, la definición de parámetros y los retornos (si aplican). El programa deberá realizarse aplicando arreglos y archivos.
 - Pruebas: ejecuta las pruebas y documenta la evidencia de cada caso de prueba. Genera el reporte de la ejecución de pruebas.

Recuerda que el programa debe usar las mejores prácticas de documentación e indentación de código.

Entregables

Documento con el desarrollo del ejercicio para describir el proceso de producción.

• • • • **Bloque 2** • • • •

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad, de preferencia física, para ayudar al alumno a reconectarse a la clase (ejemplo, sentadillas, estiramientos, etcétera).	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 14. Arreglos multidimensionales	20 minutos
Actividad del tema	Realizar la parte 2 de la Evidencia 2.	15 minutos
Cierre de bloque	El profesor deberá generar una actividad con el apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	5 minutos
Receso	Se brindará un espacio de receso para que el alumno lo utilice a su beneficio.	15 minutos


Evidencia 2
Parte 2

2. Elabora un reporte con el procedimiento que seguiste para desarrollar el software y los problemas que presentaste para lograr el correcto funcionamiento.

Entregables

Reporte que incluya la descripción de los clientes y el modelo de transporte.

Bloque 3

Actividad	Descripción	Duración
Actividad de reconexión	El profesor guiará una actividad que refuerce alguna fortaleza. Intervenciones positivas.	5 minutos
Desarrollo de temas de la semana	Revisión del tema Tema 15. Manipulación de archivos	20 minutos
Actividad del tema	Realizar la parte 3 de la Evidencia 2.	20 minutos
Cierre de la sesión	El profesor deberá generar una actividad con apoyo de herramientas tecnológicas diversas para una dinámica, quiz o foro plenario.	10 minutos



Evidencia 2

Parte 3

3. Realiza la justificación de tu diseño.

Entregable

Documento con la solución al modelo planteado y el desarrollo de la conclusión.

El resultado de la evidencia realizada durante la semana se deberá concentrar en **un solo documento**, el cual se entregará a través de la plataforma tecnológica para su revisión y evaluación por parte del docente, con base en la rúbrica de evaluación disponible en el Anexo 2.

Criterios de evaluación de la semana

Ver Anexo 2

Anexo 1. Rúbrica de evidencia 1

Competencia: Soluciona problemas utilizando un lenguaje de programación de última generación, aplicando la lógica y algoritmia en su solución.

Nivel taxonómico: Utilización

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Análisis del programa.	30 - 27	26 - 21	20 - 0	
	Muestra el análisis completo y preciso, incluyendo los siguientes elementos: datos de entrada, salida, y procesos.	Muestra el análisis correcto de manera general, incluyendo los siguientes elementos: datos de entrada, salida, y procesos.	Muestra un análisis impreciso omitiendo alguno de los siguientes elementos: datos de entrada, salida, y procesos.	
2. Diseño del programa.	30 - 27	26 - 21	20 - 0	
	1.El algoritmo es claro y se relaciona perfectamente con el problema planteado. 2. El diagrama de flujo se encuentra completo, con excelente uso de símbolos.	1. El algoritmo es confuso, pero se relaciona con el problema planteado. 2. El diagrama se muestra confuso y con ciertos elementos innecesarios o faltantes.	1. El algoritmo es confuso, pero se relaciona con el problema planteado. 2. El diagrama no tiene relación con el algoritmo.	
3. Implementación del programa.	30 - 27	26 - 21	20 - 0	
	1. Muestra un código en C# correcto. 2. Las capturas de pantalla son claras y visibles y muestran una correcta ejecución del programa.	1. Muestra un código en C# correcto. 2. Las capturas de pantalla son difícil de visualizar.	1. Muestra un código semicorrecto en C#. 2. Las capturas de pantalla muestran error en la ejecución del programa.	
3. Reporte	10 - 9	8 - 7	6 - 0	
	Presenta un reporte preciso y completo del procedimiento para lograr la solución y los problemas que se presentaron.	Presenta un reporte completo del procedimiento para lograr la solución y los problemas que se presentaron.	Presenta un reporte general del procedimiento para lograr la solución y los problemas que se presentaron.	
			TOTAL	100%

Anexo 2. Rúbrica evidencia 2

Competencia: Soluciona problemas utilizando un lenguaje de programación de última generación, aplicando la lógica y algoritmia en su solución.

Nivel taxonómico: Utilización

Criterios de evaluación	Nivel de desempeño			%
	Altamente competente 100%-86%	Competente 85%-70%	Aún sin desarrollar la competencia 69%-0%	
1. Análisis del programa.	30 - 27	26 - 21	20 - 0	
	Muestra el análisis completo y preciso, incluyendo los siguientes elementos: datos de entrada, salida, y procesos.	Muestra el análisis correcto de manera general, incluyendo los siguientes elementos: datos de entrada, salida, y procesos.	Muestra un análisis impreciso omitiendo alguno de los siguientes elementos: datos de entrada, salida, y procesos.	
2. Diseño del programa.	30 - 27	26 - 21	20 - 0	
	1.El algoritmo es claro y se relaciona perfectamente con el problema planteado. 2. El diagrama de flujo se encuentra completo, con excelente uso de símbolos.	1. El algoritmo es confuso, pero se relaciona con el problema planteado. 2. El diagrama se muestra confuso y con ciertos elementos innecesarios o faltantes.	1. El algoritmo es confuso, pero se relaciona con el problema planteado. 2. El diagrama no tiene relación con el algoritmo.	
3. Implementación del programa.	30 - 27	26 - 21	20 - 0	
	1. Muestra un código en C# correcto. 2. Las capturas de pantalla son claras y visibles y muestran una correcta ejecución del programa.	1. Muestra un código en C# correcto. 2. Las capturas de pantalla son difícil de visualizar.	1. Muestra un código semicorrecto en C#. 2. Las capturas de pantalla muestran error en la ejecución del programa.	
3. Reporte	10 - 9	8 - 7	6 - 0	

	Presenta un reporte preciso y completo del procedimiento para lograr la solución y los problemas que se presentaron.	Presenta un reporte completo del procedimiento para lograr la solución y los problemas que se presentaron.	Presenta un reporte general del procedimiento para lograr la solución y los problemas que se presentaron.	
TOTAL				100%