PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

Guía 2 Lógica de programación y Algoritmia

1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en Aplicaciones para Dispositivos Móviles
- Código del Programa de Formación 228123
- Nombre del Proyecto:
- Fase del Proyecto, Fase 3: Ejecución del Proyecto (Ejecución)
- Actividad de Proyecto:Solucionar Problemas con algoritmos y lógica de programación
- Competencias:

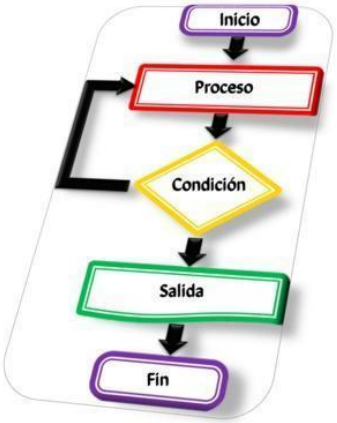
Técnicas:

- Desarrollar la solución de software de acuerdo con el diseño y metodologías de desarrollo.

- Resultados de Aprendizaje Alcanzar:
  - Identificar los componentes de un algoritmo para dar solución a un problema.
  - Codificar el algoritmo de la solución del problema de acuerdo al lenguaje de programación seleccionado
  - Diseñar algoritmos de acuerdo con el problema a solucionar.

- Duración de la Guía: 80 horas (64 horas presenciales y 16 horas desescolarizadas)

2. PRESENTACIÓN



2. Presentación

Toda persona que se inicie en el mundo de la programación necesariamente debe tener en cuenta algunos elementos esenciales para llegar a ser un buen programador. La lógica es uno de estos elementos. Ésta es esencial en el conocimiento de la programación, porque es la base para escribir código que la computadora pueda interpretar correctamente. La lógica nos permite desarrollar aplicaciones utilizando una secuencia razonada de pasos para lograr determinado objetivo.

https://lucashiguitabedoya.jimdofree.com/2-periodo-2/contenido/

Esta guía le permitirá al aprendiz introducirse en el mundo de la programación a través de una serie de conocimientos que les darán las herramientas necesarias para obtener los procesos mentales adecuados para aprender a programar. Es de vital importancia que los aprendices realicen las actividades propuestas

con conciencia, ya que esto les permitirá adquirir los conocimientos y la lógica necesaria de manera efectiva.

Los programadores no se hacen de la noche a la mañana, es un proceso que se puede ir realizando poco a poco y la lógica es uno de los aspectos que primero se deben desarrollar. Eso sí, hay que tener algunos conocimientos previos, por ejemplo, manejo básico del computador y fundamentos de matemáticas, entre otros, también hay que tener mucha curiosidad y práctica constante.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 Actividades de Reflexión inicial.

Actividad de Aprendizaje 03_1: Analizar los elementos básicos de la lógica proposicional y la construcción de algoritmos para entender el campo de aplicación de la lógica.

Tiempo estimado: 2 horas.

Descripción de la(s) actividad(es)

Antes de iniciar las actividades propuestas, conforme grupos, máximo de 2 personas y aliste una hoja de papel.

Observe del video **2.1¿QuéesPYTHONyparaquéSIRVE_.mp4 que se encuentra en la carpeta material del aprendiz** o ingrese a la dirección https://www.youtube.com/watch?v=lc5JJTQa4r8 y observe el video: que es Python y para que sirve y tener un mejor concepto hacia dónde va orientada esta guía.

Detengamos el tiempo por un segundo y recordemos cómo se elabora una figura animal de papel. Sigan los pasos que se les indican a continuación:

Parte 1:

Paso 1. Con la hoja de papel realice la figura de un animal (Mariposa, pavo real, elefante, pez, etc.).

Paso 2. Junto a sus compañeros de grupo escriba en inglés en una hoja como sería el paso a paso para elaborar la figura de un animal en papel.

Paso 3. ¡Ahora reflexionemos! ¿Qué pasaría si cambiamos los materiales o la forma del animal? ¿Cambiaría el paso a paso descrito? Escriban sus conclusiones utilizando el idioma inglés.

Parte 2:

¡Intentemos de nuevo! Con los mismos compañeros de grupo, analicen y escriban:

a) ¿Cómo sería la solución paso a paso de la siguiente operación matemática? Dado el costo de un computador portátil y la cantidad de dinero entregado por el cliente, calcule e imprima el cambio que debe devolvérsele al cliente.
b) ¿Cuáles conclusiones pueden obtener, relacionadas con el tema de la sesión de hoy, a partir de las actividades anteriores? Por favor, comenten en grupo y registren por escrito sus conclusiones.

Para finalizar la actividad en compañía del instructor, elijan a un compañero del grupo, que comparta la experiencia a los demás aprendices que se encuentran en el ambiente de formación (Parte 1 y Parte 2).

Parte 3:

a) De forma individual van a ingresar al juego Bloxorz en el siguiente enlace https://www.friv.com/z/games/bloxorz/game.html y van a realizar una serie de pasos para ir avanzando de nivel
   1 ¿En cuánto tiempo y serie de pasos realizo para avanzar al nivel 2?
   2. ¿Qué nivel fue complicado? ¿Qué nivel se desarrolló en menos tiempo?

b) Instrucciones del juego de Nim:
   1. Seleccionen a dos personas para jugar.
   2. Dibujen en el tablero del salón de clases 12 óvalos formando un círculo.
   3. Por turnos, cada jugador deberá borrar uno o dos óvalos del círculo.
   4. Recuerden que solo se pueden borrar óvalos que estén juntos en el círculo y no se pueden saltar fichas.
   5. El objetivo del juego es ser el jugador que borre el último óvalo.
   6. Jueguen varias partidas e intercambien los roles para que todos tengan la oportunidad de jugar y analizar las estrategias.
   7. Observen y discutan juntos para identificar la estrategia ganadora, es decir, la mejor manera de jugar para asegurarse de ser quien borre el último óvalo.
   8. Diviértanse mientras ponen en práctica su lógica y estrategia en este desafiante juego de Nim en el tablero del salón de clases.

¡Que empiece la diversión y que el mejor estratega gane en el tablero del salón de clases!

Referencia del juego de Nim: Sáenz de Cabezón, E. (11 de julio de 2018). Las matemáticas nos hacen más libres y menos manipulables [Video]. Aprendemos Juntos 2030. Recuperado el 10 de julio de 2023, de https://youtu.be/BbA5dpS4Ccl?t=1587 (26:27).

Indicaciones para el entregable o evidencia: Crear un documento en formato PDF que contenga las respuestas a las preguntas de la Parte 1 y Parte 2 de las actividades de reflexión inicial. Guardar como **02_1_ActividadReflexiónInicial_LógicaDeProgramación_TuapellidoTunombre.pdf**. Recuerda seguir estas instrucciones para organizar y presentar adecuadamente tu entregable o evidencia.

Ambiente requerido: Aula de clase.

Materiales: Hojas de papel, lápices, esferos, tablero, marcadores, computadores, VideoBeam.

3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje

3.2.1 Actividad de Aprendizaje 06_2 Identificar los conocimientos necesarios para elaborar algoritmos básicos.

Tiempo estimado: 6 Horas.

Siga las indicaciones que se presentan a continuación:

Parte 1:

1. Haz clic en el siguiente enlace para acceder al juego "Voyage Aquatic – Code.org": https://studio.code.org/s/aquatic/lessons/1/levels/1. Sumérgete en esta aventura submarina y desarrolla tu lógica de programación mientras te diviertes.
2. **Objetivo:** Explora, resuelve desafíos y avanza en el juego utilizando la programación como herramienta principal.
3. **Dinámica:**
   - Juego individual: Supera los desafíos por tu cuenta y descubre nuevas formas de aplicar la lógica de programación en el entorno submarino.
   - Juego en parejas: Formen equipos y asignen roles. Uno controla el personaje y el otro analiza y escribe el código necesario.
4. **Comunicación y colaboración:**
   - ¡Trabajen juntos! Comuníquense, compartan ideas y tomen decisiones conjuntas para resolver los desafíos.
5. **Turnos de juego (solo para parejas):**
   - Alterna el control del personaje y la escritura de código para que ambos participen activamente.

   ¡Sumérgete en "Voyage Aquatic" y disfruta de esta experiencia desafiante y divertida mientras desarrollas tu lógica de programación!
6. A continuación, observe el siguiente video: 05_VID_QuéEsUnAlgoritmo.mp4 que se encuentra en el material de apoyo del aprendiz.
7. Comparte y socializa con tus compañeros si este último juego tiene relación con el video visto.

Parte 2:

¡Bienvenidos de vuelta! En actividades anteriores, hemos tenido la oportunidad de jugar juegos que otros han creado. Ahora, es el momento de cambiar de rol y poner en práctica nuestra creatividad y habilidades de programación por bloques para crear tu propio juego, un juego de laberinto.

1. ¡Accede a la plataforma MIT App Inventor en https://appinventor.mit.edu/ y dale clic en la opción “Create Apps!” (¡Crear aplicaciones!).
2. Observa el video **2_2COMOCREARUN LABERINTOEN MITApInventor 2.mp4** que se encuentra en la carpeta material del aprendiz para poder crear la aplicación app, sigue los pasos

Recuerda aprovechar los recursos y documentación disponibles en MIT App Inventor para ayudarte en el proceso de desarrollo.

¡Disfruta creando tu juego de laberinto en MIT App Inventor!

Indicaciones para el entregable o evidencia: Crear un documento en formato PDF que contenga las capturas de pantalla de los juegos de la Parte 1 y Parte 2. Guardar como **02_2_ActividadContextualización _LógicaDeProgramación_TuapellidoTunombre.pdf**. Recuerda seguir estas instrucciones para organizar y presentar adecuadamente tu entregable o evidencia.

3.2.2 Actividad de Aprendizaje 05_2 Identificar los conocimientos básicos necesarios para elaborar proposiciones simples y compuestas

Tiempo estimado: 4 Horas.

1. Observen el video 02_VID_ProposicionesYOperacionesBásicas.mp4 que se encuentra en el material de apoyo del aprendiz.
2. En grupos de dos personas realicen la lectura del documento denominado 05_LECT_IntroducciónALaLógica.pdf que se encuentra en el material de apoyo del aprendiz.
3. Elaboren en su cuaderno 10 enunciados diferentes (en español y en inglés) y traten de observar si son o no proposiciones. Clasifíquelas en compuestas o simples al frente de cada enunciado.
4. Socialicen (en español y en inglés) con los demás grupos los enunciados trabajados y los resultados obtenidos.

3.2.3 Actividad de Aprendizaje: 06_2 Identificar los conocimientos básicos necesarios para manipular las entidades primitivas en programación como paso previo para iniciar el proceso de creación de algoritmos en Python

Tiempo estimado: 6 Horas.

La siguiente actividad se debe realizar en grupos de 2 aprendices.

1. Realicen la siguiente lectura 02_LECT_EntidadesPrimitivas.pdf.
2. Observar el siguiente video 02_VID_ResoluciónDeExpresiones_ (Lógica De La Programación).mp4.
3. Descargar y resolver los ejercicios del taller 2_1_TALL_ExpresionesMatemáticas.docx aplicando los temas de la lectura y para el entregable o evidencia guardar como

   2_1_1_TALL_ExpresionesMatemáticas _TuapellidoTunombre.pdf

3.2.4 Actividad de Aprendizaje 02_2 Identificar los conocimientos necesarios para elaborar algoritmos básicos.

Tiempo estimado: 7 Horas.

Realizar la siguiente actividad en español e inglés, deberá seguir las indicaciones que se presentan a continuación:

1. Conformar grupos de trabajo, máximo 3 personas.
2. Refresque su conocimiento sobre el tema observando nuevamente el video 02_VID_QuéEsUnAlgoritmo.mp4 que ya ha sido previamente visto.

3. También observe el siguiente video 02_VID_LógicaDeProgramaciónEstructuraSelectiva.mp4.
4. A continuación, en cualquier aplicativo dibujen las figuras que se muestran a continuación:

Nombre:_____ 	Nombre:_____ 	 Nombre:_____	 Nombre:_____	 Nombre:_____

5. Luego, basándose en los videos vistos, identifiquen cada una de las figuras y describan su utilidad o funcionamiento. Indiquen en qué parte de un diagrama de flujo se pueden utilizar.
6. Realicen la lectura del siguiente material de apoyo que se encuentra en la carpeta material del aprendiz: **2_Instrucciones Básicas en un Algoritmo en python.pdf**.
7. A continuación, con el instructor desarrollara los dos ejercicios propuestos en Python en el archivo denominado **5_1Instrucciones_Basicas_Allgoritmo_python.pdf** que se encuentra en la carpeta material del aprendiz

Ejemplo:

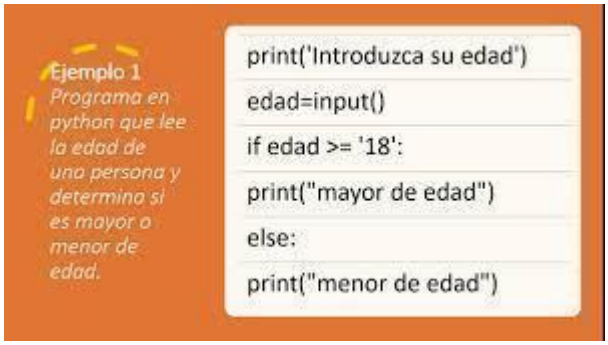


Ilustración 2: Diagrama de flujo tomado de https://www.google.com/search?q=Instrcucciones+Basicas+de+un+Algoritmo+en+python&sca_esv=d7e7d90bd6b3a957&udm=2&biw=1358&bih=608&ei=7aNZZ8vBBfSWwbkP8pOAwAc&ved=0ahUKEwiL7MHs-J-KAxV0SzABHfIJAHgQ4dUDCBE&uact=5&oq=Instrcucciones+Basicas+de+un+Algoritmo+en+python&gs_lp=EgNpbWciMEluc3RyY3VjY2lvbmVzIEJhc2ljYXMgZGUgdW4gQWxnb3JpdG1vIGVuIHB5dGhvbkivLFDgBVj_J3ABeACQAQCYAcoGoAG1F6oBCzAuMy41LjEuNi0xuAEDyAEA-AEBmAIAoAIAmAMA4gMFEgExIECIBgGSBwCgB8ID&sclient=img#vhid=pF7mdcXqTvrIOM&vssid=mosaic

Hasta aquí analizamos problemas y cómo podemos darles la solución de diversas maneras. A continuación, nos enfocaremos en situaciones que involucran enunciados relacionados con la programación, como el manejo de información sobre sueldos, personas, calificaciones, transacciones bancarias y más.

8. Visualizar el siguiente video **2.EstructurasselectivasPython3.mp4 ,** realizar el ejercicio en Python y luego en la carpeta material del aprendiz colocar en práctica desde python el ejercicio que se encuentra en el archivo denominado **prueba_conocimiento.pdf.**
9. Para el entregable o evidencia guardar en su portafolio de evidencia y aplicativo sena como **ejercicios_basicos_python.pdf**

Ambiente requerido: Aula de clase.

Materiales: Video beam, pantalla tv, cable HDMI, cable VGA, guía de formación, marcadores, resma de papel, tablero, conexión a internet, equipos de cómputo Visual Studio Code

3.3 Actividades de apropiación del conocimiento (Conceptualización y Teorización).

3.3.1 Actividad de Aprendizaje 06_3 Construir tablas de verdad aplicando los conocimientos adquiridos para entender las respuestas lógicas obtenidas con el computador.

Tiempo estimado: 5 horas.

La actividad se debe realizar en grupos de 2 personas.

1. Observar el video 02_VID_TablasDeVerdadProposicionesCompuestas.mp4 contenido en la carpeta Material de Apoyo.
2. Leer el siguiente material de apoyo 05_LECT_IntroducciónALaLógica.pdf.
3. En compañía del instructor realicen el taller que encontrarán en el documento denominado 2_2_TALL_TablasDeVerdad.docx, el cual les permitirá aplicar los conocimientos adquiridos durante la sesión y para el entregable o evidencia guardar como **2_2_TALL_TablasDeVerdad_TuapellidoTunombre.pdf**.

3.3.2 Actividad de Aprendizaje 06_3 Emplear estructuras de programación secuenciales para la resolución de problemas

Tiempo estimado: 5 horas.

En grupos de máximo 3 personas realicen la siguiente actividad:

1. Observar el siguiente video 02_VID_QuéEsUnAlgoritmo.mp4.
2. Leer el documento 02_LECT_FundamentosYEstructurasSecuenciales.pdf, que se encuentra en el material de apoyo del aprendiz.
3. Revisar los ejemplos.
4. Realizar los ejercicios en el documento 2_3_TALL_FundamentosYEstructurasSecuenciales_en python.docx, que se encuentra en el material de apoyo del aprendiz y para el entregable o evidencia guardar como 2**_3_TALL_FundamentosYEstructurasSecuenciales_TuapellidoTunombre.docx**.

3.3.3 Actividad de Aprendizaje 06_3 Emplear estructuras de selección para crear algoritmos que puedan cambiar la secuencia de un proceso.

Tiempo estimado**:** 6 horas.

En grupos de máximo 3 personas realicen la siguiente actividad:

1. Observar el video 02_VID_EstructurasCondicionalesEnAlgoritmos.mp4.
2. Leer el documento 02_LECT_EstructurasDeSelección.pdf. con la explicación del instructor realizar los ejercicios que se encuentran en el documento.

3. Descargar y realizar el taller del archivo **2_4_TALL_FundamentosYEstructurasSelectivas_python.docx** y para el entregable o evidencia guardar como **64_TALL_FundamentosYEstructurasSelectivas_TuapellidoTunombre.pdf**.

3.3.4 Actividad de Aprendizaje 06_3 Emplear estructuras iterativas para crear algoritmos que puedan cambiar la secuencia de un proceso.

Tiempo estimado: 6 horas.

En grupos de máximo 2 personas realicen la siguiente actividad:

1. Leer el documento 02_LECT_EstructurasDeRepetición.pdf, revisar y analizar detenidamente los ejemplos que allí se encuentran.
2. Ingresar a la siguiente página y estudiar las lecciones 1, 2 y 3:

   http://aprende.colombiaaprende.edu.co/sites/default/files/naspublic/curriculos_ex/n1g10_fproy/nivel1/programacion/unidad4/leccion1.html

3. Descargar y realizar el taller 2_5_TALL_FundamentosYEstructurasIterativas.docx y para el entregable o evidencia guardar como 2**_5_TALL_FundamentosYEstructurasIterativas_TuapellidoTunombre.pdf**.

3.3.5 Actividad de Aprendizaje 06_3 Cuestionario Lógica de Programación y Algoritmia para validar los conocimientos adquiridos por los aprendices.

Tiempo estimado: 1 hora.

En esta actividad se debe resolver el cuestionario de la Guía 2 que se encuentra en Zajuna. Para ello realice los siguientes pasos:

1. Ingrese a Zajuna.
2. Haga clic en el elemento con el icono del SENA y de nombre Programación de software.
3. En el panel de la izquierda haga clic en la opción Contenido del curso.
4. Haga clic en la carpeta actividades del proyecto. Y a continuación, haga clic en fase 3_Ejecucion.
5. luego haga clic en la carpeta competencia.
6. Haga clic en la Carpeta Guía 5 - Lógica de programación y algoritmia.
7. En esta carpeta encontrará un documento con el nombre de Cuestionario lógica programación Guía 5, haga clic en él. Y para empezar a desarrollar el cuestionario haga clic en el botón Continuar.
8. Al terminar haga clic en el botón enviar.

Ambiente requerido: Aula de informática.

Materiales: Video beam, pantalla tv, cable HDMI, cable VGA, guía de formación, marcadores, resma de papel, tablero, conexión a internet, equipos de cómputo, Visual Studio Code

3.4 Actividades de transferencia del conocimiento

Actividad de Aprendizaje 06_4: Aplicar los conceptos de Lógica de Programación y algoritmia en el proyecto seleccionado.

Tiempo estimado: 32 horas (25 presenciales, 7 desescolarizadas)

Por grupos de proyecto diseñe los diagramas de flujo y el pseudocódigo del módulo Usuarios en python que tiene el proyecto asignado manejando el registro, consulta y actualización de los datos ingresados.

Para el registro de un usuario el algoritmo debe pedir los datos del usuario a registrar, por ejemplo, nombre de usuario y contraseña y los demás datos necesarios, según lo indicado en el archivo plantilla del proyecto asignado.

Para la consulta de usuarios el algoritmo debe permitir consultar los usuarios según su número de documento de identificación.

Igualmente, para la actualización de los datos de un usuario, el algoritmo debe pedir el número de identificación del usuario al cual se le actualizarán los datos y luego se debe pedir el dato a actualizar.

Para la entrega de la evidencia, les solicitamos que coloquen los diagramas de flujo y pseudocódigos de su proyecto en un documento de Word y lo guarden como **02_InformeAplicaciónAlgoritmosProyecto_TuapellidoTunombre.pdf**.

Ambiente requerido: Aula de informática.

Materiales: Video beam, pantalla tv, cable HDMI, cable VGA, guía de formación, marcadores, tablero, conexión a internet, equipos de cómputo, Visual Studio Code.

4. Actividades de evaluación

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento: ● 02_3 Cuestionario Lógica de Programación y Algoritmia	● Da respuesta sobre tablas de verdad, conectores lógicos, entidades primitivas, algoritmos, fases de los algoritmos, estructuras y su aplicación en la solución de un problema.	Prueba de conocimientos cuestionario Zajuna
Evidencias de Desempeño: 2_1TALL_ExpresionesMatemáticas.pdf 2_2_TALL_TablasDeVerdad.pdf	● Elabora Tablas de verdad de acuerdo con los problemas planteados. ● Representa procesos del sistema a partir de la construcción de	Observación Sistemática / Lista de Chequeo.

2_3_TALL_FundamentosYEstructurasSecuenciales.pdf 2_4_TALL_FundamentosYEstructurasSelectivas.pdf 2_5_TALL_FundamentosYEstructurasIterativas.pdf	• algoritmos, como parte de la solución a situaciones planteadas, utilizando lenguajes de programación orientados a objetos. • Realizar algoritmos utilizando variables, constantes, variables, bucles, contadores, acumuladores, selección, seudocódigo.	
Evidencias de Producto: 02_InformeAplicaciónAlgoritmosProyecto.docx		Valoración de Producto Rubrica.

4. Glosario de términos

Lectura: Consiste en recibir desde un dispositivo de entrada (p.ej. el teclado) un valor.

Escritura: Consiste en mandar por un dispositivo de salida (p.ej. monitor o impresora) un resultado o mensaje.

Programación: Es la implementación de un algoritmo en un determinado lenguaje

de programación, para realizar un programa.

Algoritmo: Secuencia de pasos lógicos para dar solución a un problema, con un inicio y fin definidos.

Pseudocódigo: Es una descripción de alto nivel de un algoritmo que emplea una mezcla de lenguaje natural con algunas convenciones sintácticas propias de lenguajes de programación, como asignaciones, ciclos y condicionales.

Programa: (Software en inglés) es una secuencia de instrucciones que una

computadora puede interpretar y ejecutar.

Variable: Espacio reservado de memoria cuyo valor puede cambiar durante la ejecución del programa.

Constante: Espacio reservado de memoria que permanece constante durante la ejecución del programa.

Diagrama de Flujo de Datos: Representación gráfica de un algoritmo.

Asignar: Establecer un valor a una variable.

Condicional: Estructura algorítmica de decisión que permite seguir un camino verdadero o falso en base a la o las condiciones.

Ciclo Anidado: Estructura algorítmica que consiste en ubicar un ciclo dentro de otro para realizar alguna tarea específica.

Índex: Variable de tipo entera encargada de controlar (contar) el número de iteraciones o veces que se repite un ciclo.

Ciclo: Estructura algorítmica que permite repetir una o varias líneas de código según se cumpla una condición.

Valor de verdad: Unidades para validación de los procesos de la lógica, puede ser Verdadero o Falso ó en su defecto Si o No ó 1 o 0.

6. Referentes bibliográficos

- Colombia aprende (sf). Currículos exploratorios en TIC. http://aprende.colombiaaprende.edu.co/sites/default/files/naspublic/curriculos_ex/n1g10_fproy/nivel1/programacion/unidad4/leccion1.html
- Instituto tecnológico de Sonora (2016). Pseudocódigo y PSEINT. https://www.itson.mx/oferta/isw/Documents/guia_pseint_2016.pdf
- Lógica matemática. Unidad 1. https://virtual.uptc.edu.co/ova/cursos/logica/unidad_uno/logica_unidad1.pdf

Prieto, S. (sf). Principios de algoritmia. https://docplayer.es/42338535-Principios-de-algoritmia.html

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	**Jannez Urrego**	Instructor	Articulación con la media	13/12/2021
Autor (es)	Maria Angelica Triviño	Instructor	Articulación con la media	02 de febrero 2017
Autor (es)	Giovanni Urrego Herrera	Instructor	Articulación con la media	15 de febrero de 2019
Autor (es)	Wilmer Cardozo Lopez Jennifer Fajardo	Instructor	Articulación con la media	08 de octubre de 2019
Autor (es)	Kevin García Aguilar	Instructor	Articulación con la media	05/12/2019
Autor (es)	Mauricio Lara r.	Instructor	Articulación con la media	30/11/2020

Autor (es)	Luis Felipe Martinez Guerrero	Instructor	Articulación con la media	13/12/2021

8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)	Fabian Hernández	Instructor	Articulación con la media	03/07/2023	Actualización del diseño curricular.
Autor (es)	William Ramon Flórez	Instructor	Articulación con la media	11/12/2024	Actualización del diseño curricular.
Autor (es)	Joan Reyes Atencio	Instructor	Articulación con la media	11/12/2024	Actualización del diseño curricular.