





## 2. Verificación del ambiente de formación

Realicé la verificación del ambiente, con el fin de impartir la Formación en el Técnico en programación de software.

### Aspectos Revisados:

- ✓ Correcta iluminación.
- ✓ Condiciones ambientales.
- ✓ Riesgos eléctricos.

Se encuentra **APTO**, para impartir la formación.

## 3. Desarrollo de la sesión de formación:

### ➤ Acuerdo de normas para el desarrollo de la sesión de formación:

- ✓ Pedimos la palabra levantando la mano.
- ✓ Participación y presencia plena en el ambiente de aprendizaje.

### ➤ Verificación de la asistencia mediante el diligenciamiento Registro de Asistencia formato GOR-F-085 V02, se informa sobre el contenido de la Ley 1581 de 2012 Protección y tratamiento de datos.

### ➤ ¿Reflexión grupal (Activador Cognitivo): se comparte un mensaje de reflexión antes de iniciar la formación con el objetivo de aplicarlos en los procesos formación y el quehacer diario de la vida cotidiana?

➤ Recomendaciones para mantener actualizado la información de los aprendices en <http://senasofiaplus.edu.co>, con el fin de evitar inconvenientes en un futuro en la certificación de los aprendices que aprueben el proceso formativo. Se requiere de su colaboración para verificar, corregir y actualizar el registro. Es de recordar que el aprendiz cuando cumple la mayoría de edad y se encuentra en formación debe aportar al instructor el comprobante de la cedula para proceder a realizar el cambio en la plataforma Sofiaplus.

### ➤ Refuerzo sobre el reglamento del Aprendiz

#### **Capítulo III: Deberes del aprendiz SENA**

En este capítulo se explican los deberes del aprendiz SENA del reglamento del aprendiz SENA:

- Cumplir con todas las actividades propias de su proceso de aprendizaje o plan de mejoramiento definidos durante la etapa lectiva y productivo
- Respetar los derechos ajenos y no abusar de los propios
- Acatar las decisiones del Manual de convivencia
- Asumir con responsabilidad su participación en las actividades programadas
- Respetar los derechos de autor en los materiales que hayan sido compartidos.
- Conocer y asumir las políticas y directrices establecidas, así como el Reglamento del Aprendiz SENA
- Mantener en buen estado, orden y aseo las instalaciones físicas
- Actuar según principios y valores para la convivencia
- Informar al instructor, coordinador, directivo o personal de apoyo de cualquier irregularidad
- No usar expresiones grotescas en foros de discusión, mensajes, anuncios... Además de lo explicado anteriormente, el aprendiz considerará también las siguientes obligaciones:
- ✓ Acceder regularmente a los ambientes virtuales de formación



- ✓ Dedicar al proceso de aprendizaje las horas semanales señaladas para cada programa
- ✓ Ingresar a la plataforma virtual debidamente identificado con el código de acceso personal e intransferible
- ✓ Realizar personalmente las evaluaciones, investigaciones y actividades de formación
- ✓ Participar en los foros de discusión
- ✓ Respetar los comentarios realizados por los demás integrantes
- ✓ Cumplir con los trabajos indicados por los tutores

#### **Capítulo IV: Prohibiciones del reglamento del aprendiz SENA**

Las prohibiciones especificadas del reglamento del aprendiz SENA son las siguientes:

- Plagiar materiales, trabajos y otros documentos
- Terminar unilateralmente el contrato de aprendizaje
- Realizar fraude en evaluaciones
- Aportar documentación que difiera con la real
- Fumar en áreas no permitidas
- Ingresar o portar armas
- Utilizar el nombre del SENA para actividades particulares o con ánimo de lucro
- Hurtar, estafar o abusar de la confianza de cualquier integrante de la comunidad educativa
- Contribuir al desorden y/o desaseo
- Destruir o dañar instalaciones físicas, equipos, software...
- Obstaculizar el ingreso a las instalaciones
- Practicar juego de azar
- Realizar acciones proselitistas
- Permanecer con el uniforme acordado en situaciones que deteriore la imagen institucional
- Generar, transmitir o publicar información violenta, insultos, ...
- Actos que saboteen actividades de formación
- Suplantar la identidad
- Falsificar documentos oficiales

#### **➤ Actividades realizadas durante la sesión:**

#### **Objetivos de la formación:**

- Impartir la formación profesional integral.
- Desarrollar la guía de aprendizaje.
- Realizar las actividades de desempeño.
- Presentar por parte del aprendiz las actividades de conocimiento.
- Realizar lecturas al material de apoyo para entrega de las evidencias producto de la formación impartida por el instructor.

#### **Introducción:**

Se realiza una breve introducción de la competencia y los resultados de aprendizaje que se presenten alcanzar y de cada una de las actividades de aprendizaje que el aprendiz debe realizar para aprobar la competencia antes mencionada. Se recuerda que las calificaciones en el Sena son D que no es competente y A que es competente.

**Revisión de conceptos previos:** El instructor realiza una breve Antes de profundizar en el estudio de la programación de software, es esencial comprender una serie de conceptos fundamentales que sirven como base para el desarrollo de aplicaciones y sistemas informáticos. Estos conceptos permiten



entender cómo se estructuran, diseñan y ejecutan las soluciones computacionales, facilitando la interpretación de los lenguajes de programación, la lógica algorítmica y las metodologías de desarrollo. Al revisar términos clave como algoritmos, variables, estructuras de control, funciones y paradigmas de programación, se establece un marco de referencia claro que permite al lector o aprendiz avanzar con mayor seguridad en el proceso de análisis y construcción de software. Esta comprensión previa es indispensable para garantizar un aprendizaje sólido y coherente en el área del desarrollo de software.

**Situaciones problema:** El estudio de los algoritmos y la programación de software no solo permite entender cómo funcionan los programas, sino también optimizar recursos, mejorar el rendimiento de las aplicaciones y garantizar resultados precisos. En la lógica de programación, el diseño de algoritmos es el primer paso para transformar una necesidad o problema real en una solución digital. Por ello, conocer sus elementos, tipos y formas de representación es esencial para todo programador o desarrollador de software.

**Contextualización:** La importancia de representar algoritmos con pseudocódigo radica en su capacidad para acercar al futuro programador con el código de los lenguajes de programación que utilizará para la construcción de aplicaciones, visualizar la lógica de un proceso de manera clara y concisa, facilitando la comprensión, comunicación y diseño de soluciones a problemas. Los diagramas de flujo son herramientas visuales que permiten representar gráficamente la secuencia de pasos necesarios para ejecutar una tarea, identificar posibles errores o mejoras, y optimizar la eficiencia de un algoritmo.

Los algoritmos son secuencias de pasos lógicos, finitos y ordenados que permiten resolver problemas o realizar tareas específicas. Aunque su uso es milenario y sus fundamentos pueden encontrarse en la lógica matemática y la aritmética clásica, como los aportes del matemático persa Al-Juarismi en el siglo IX, su aplicación actual ha cobrado un nuevo significado gracias al auge de la informática. Hoy en día, los algoritmos no solo se utilizan para resolver cálculos matemáticos, sino también para tomar decisiones, analizar grandes volúmenes de datos, predecir comportamientos, automatizar procesos y optimizar recursos en diversas áreas como la medicina, el comercio, la industria, la educación, la seguridad y el entretenimiento.

Por ejemplo, cuando realizamos una búsqueda en Google, el sistema utiliza algoritmos de indexación, clasificación y relevancia para mostrar los resultados más adecuados en milisegundos. Del mismo modo, las redes sociales utilizan algoritmos de recomendación para mostrar publicaciones y anuncios personalizados, basándose en nuestro comportamiento digital. Incluso en campos críticos como la medicina, los algoritmos ayudan a diagnosticar enfermedades mediante inteligencia artificial, a partir del análisis de imágenes o patrones clínicos. Esto demuestra que su influencia no es menor: los algoritmos están moldeando la forma en que interactuamos con el mundo.

El desarrollo de software es la disciplina encargada de traducir ideas o necesidades humanas en soluciones digitales funcionales mediante lenguajes de programación. Este proceso va más allá de simplemente escribir líneas de código: implica identificar problemas, analizar requerimientos, diseñar soluciones, estructurar algoritmos, codificar, probar, depurar, documentar y mantener los sistemas a lo largo del tiempo. La programación de software es, por tanto, una habilidad técnica y cognitiva que exige tanta creatividad como rigor lógico. Su aprendizaje promueve el pensamiento computacional, que es la capacidad de descomponer problemas complejos, identificar patrones, abstraer soluciones y construir



algoritmos eficientes. Dominar esta competencia no solo permite crear programas útiles, sino también comprender el funcionamiento de las tecnologías que usamos cotidianamente.

La programación se ha diversificado en múltiples paradigmas como la programación estructurada, orientada a objetos, funcional o lógica, cada uno con enfoques distintos para resolver problemas. Además, existen cientos de lenguajes de programación como Python, Java, C++, JavaScript o Swift, cada uno diseñado con propósitos específicos. Sin embargo, sin importar el lenguaje, todos comparten un mismo principio: la construcción de algoritmos precisos que resuelvan un problema de manera clara y eficiente.

El impacto de los algoritmos y la programación de software trasciende el ámbito técnico. En el contexto educativo, su enseñanza fomenta habilidades clave del siglo XXI como la resolución de problemas, la creatividad, la lógica y el pensamiento crítico. En las instituciones, el desarrollo de software permite digitalizar procesos, mejorar la eficiencia, reducir errores y tomar decisiones más acertadas a partir de datos. En el entorno económico, la industria del software es uno de los sectores de mayor crecimiento y demanda de talento, con aplicaciones en sectores como el financiero, logístico, gubernamental, industrial, agrícola y cultural. Al mismo tiempo, el uso masivo de algoritmos ha despertado preocupaciones éticas y sociales. El hecho de que estos procesos automáticos tomen decisiones sin intervención humana directa —como filtrar información, aprobar créditos, gestionar recursos o detectar comportamientos— obliga a pensar en su transparencia, imparcialidad, privacidad y responsabilidad. Por ello, el estudio de algoritmos y programación no debe limitarse solo al plano técnico, sino también incluir una reflexión crítica sobre su impacto en la sociedad.

En la formación técnica, tecnológica y universitaria, los cursos de lógica de programación y diseño de algoritmos son fundamentales, ya que preparan a los estudiantes para enfrentar los retos del desarrollo de software desde una base sólida. A través del análisis de problemas reales y su resolución mediante algoritmos, los futuros programadores aprenden a estructurar soluciones que luego serán traducidas en código. Este enfoque no solo mejora sus competencias técnicas, sino que también fortalece su capacidad para trabajar en equipo, documentar procesos y enfrentar desafíos complejos. En un entorno educativo donde se promueve la formación por competencias, el desarrollo del pensamiento lógico y algorítmico representa un eje transversal en carreras de ingeniería, ciencias computacionales, mecatrónica, diseño de software y muchas más. Además, las metodologías activas como la programación por proyectos, el aprendizaje basado en problemas y el uso de plataformas interactivas permiten que el proceso de aprendizaje sea más significativo y contextualizado.

La relación entre algoritmos y programación de software es inseparable, ya que ambos constituyen la base sobre la cual se construyen las soluciones digitales que transforman nuestro entorno. En un mundo cada vez más orientado hacia la digitalización, el dominio de estas competencias no solo representa una ventaja técnica, sino también una herramienta para comprender, participar y liderar los cambios tecnológicos de forma responsable y crítica. Contextualizar su estudio implica reconocer su papel central en la vida moderna y asumir el compromiso de utilizarlos para el bien común, promoviendo la innovación con ética, inclusión y sostenibilidad.

#### **Introducción del tema:**

Un algoritmo es una **secuencia de pasos definidos y finitos que realiza una tarea o resuelve un problema en particular**. Los algoritmos son esenciales en la informática y la programación, ya que proporcionan una forma sistemática de abordar problemas y realizar cálculos.



## Desarrollo

En esta formación se evaluará los temas de programación con HTML y CSS. La evaluación es la siguiente:

A screenshot of a Google Docs document titled "GFPI-F-135FormatoGuíadeAprendizaje\_Programacion". The document content is as follows:

**¿Cuáles son los tipos de Operadores JS?**

Al trabajar con JavaScript (o con cualquier lenguaje de programación), es muy habitual hacer uso de los llamados operadores. Se trata de unos símbolos que nos permitirán hacer una serie de operaciones rápidas con uno o más operandos (generalmente números, aunque también pueden ser otros tipos de datos).

Sin embargo, esto se entiende mejor con ejemplos, por lo que vamos a hacer un desglose de los operadores que utilizaremos:

Tipo de operador	Descripción
Operadores básicos	Operadores aritméticos, de asignación y unarios.
Operadores de comparación	Permiten realizar comparaciones entre varios elementos.
Operadores binarios	Permiten hacer operaciones a bajo nivel.
Nullish Coalescing	Devuelve información si una variable es null o undefined.
Optional chaining	Realiza una operación sólo si un objeto tiene una propiedad.

GFPI-F-135 V04

The screenshot also shows the Google Docs interface with various toolbars and a Windows taskbar at the bottom with the date 29/10/2023.



### Operadores lógicos

Permiten hacer comparaciones lógicas sobre uno o varios elementos.

```
var edad = 17;  
  
//Condional para saber si es mayor de edad  
if(edad > 18 ){  
    alert("Ya puede solicitar la cédula");  
}
```

vamos a ver los operadores aritméticos, de asignación y unarios.

Nombre	Operador	Descripción
Suma	$a + b$	Suma el valor de <b>a</b> al valor de <b>b</b> .
Resta	$a - b$	Resta el valor de <b>b</b> al valor de <b>a</b> .
Multiplicación	$a * b$	Multiplica el valor de <b>a</b> por el valor de <b>b</b> .
División	$a / b$	Divide el valor de <b>a</b> entre el valor de <b>b</b> .
Módulo	$a \% b$	Devuelve el resto de la división de <b>a</b> entre <b>b</b> .
Exponenciación	$a ** b$	Eleva <b>a</b> la potencia de <b>b</b> , es decir, $a^b$ . Equivalente a <code>Math.pow(a, b)</code> .

### ¿Cuáles son las Estructuras de Control JS?

Cuando se hace un programa en JavaScript las líneas de código se ejecutan de manera secuencial; es decir, primera, segunda, tercera, cuarta, etc., pero nunca cuarta y luego tercera, siempre se enumeran de arriba hacia abajo correspondiendo esto con el flujo del programa; durante este flujo en algún punto de







## ¿Cuáles son las Estructuras de Control JS?

Cuando se hace un programa en JavaScript las líneas de código se ejecutan de manera secuencial; es decir, primera, segunda, tercera, cuarta, etc., pero nunca cuarta y luego tercera, siempre se enumeran de arriba hacia abajo correspondiendo esto con el flujo del programa; durante este flujo en algún punto de las secuencias es necesario tomar una decisión y es ahí donde se debe utilizar el concepto de condicionales con respuesta positiva o negativa, en otro punto del programa se necesitará realizar el recorrido a estructuras de datos; por ejemplo un arreglo y para ello utilizaremos el concepto de bucle.



El condicional *if*, el más conocido de los condicionales, si se cumple la condición continúa ese camino. Se escribe entre paréntesis después de la palabra reservada *if*.

GFPI







**Condicional *if/else***, este condicional indica el camino alternativo que puede seguir el programa en el caso que no se cumpla la condición de la palabra reservada *if*.

```
var edad=17;
var mensaje;

//Condicional para saber si es mayor de edad
if(edad > 18 ){

    mensaje="Ya puede solicitar la cédula";

}else{

    mensaje="Aún eres menor de edad";

}

alert(mensaje);
```

**Operador ternario**, es una forma de escribir el condicional *if/else* pero de una manera más corta; se identifica con el símbolo ?

```
var edad=17;
//Operador ternario: (condición ? Verdadero : Falso)
var mensaje = edad>18 ? 'Cédula' : 'Tarjeta
Identidad';
```



**Condicional *switch***, define casos específicos a realizar en el caso de que la variable expuesta como condición sea igual a los valores que se especifican mediante los *case*.

```
var nota = 7;
switch (nota) {
  case 10:
    calificacion = "Sobresaliente";
    break;
  case 9:
    calificacion = "Notable";
    break;
  case 7:
    calificacion = "Bien";
    break;
  case 5:
    calificacion = "Insuficiente";
    break;
  default:
    // Cualquier otro caso
    calificacion = "Nota errónea";
    break;
}
alert("Resultado" + calificacion);
```



**Bucles e iteraciones**, útiles para programar tareas que se convierten en repetitivas en donde se involucra gran cantidad de volumen de datos permitiendo su automatización, para realizar un bucle es necesario tener en cuenta los siguientes conceptos: la condición tal como lo vimos en los *if*, esta se evalúa repetidamente, la iteración es la cantidad de repeticiones que ejecutó la condición, el contador es la variable que guarda en memoria el número en el cual va la repetición, el incremento es el número que se adiciona una vez se termine un ciclo y bucle infinito cuando no se realiza el incremento al contador, esto es necesario evitarlo.

Una culminado el tiempo destinado a la evaluación, se procedió a su respectiva solución y aclaración de dudas e inquietudes al respecto.

### Evaluación

**Revisión de los Objetivos de Aprendizaje:** revisión del material de apoyo y las actividades de aprendizaje que el aprendiz realiza en la formación.

**Evaluación formativa:** Esta se ve evidencia en la entrega del producto final, que es el proyecto formativo que deben realizar los aprendices.

### Reflexión Final:

- **Observaciones durante la formación:** No se presentó ninguna novedad con los aprendices, ni con el ambiente de formación.



- **Relación de aprendices en formación:** hoy en día asistieron 18 aprendices.

N°	Número de Documento	Nombre	Apellidos
1	1110174234	NIRY SARAY	ALTURO TAPIERO
2	1030283287	ANGIE MILENA	ARIAS TIQUE
3	1111262977	MARIA DEL CARMEN	BARRAGAN OYUELA
4	1025537440	DANIELA ALEJANDRA	CAMPOS SARMIENTO
5	1197463549	JAIRD SANTIAGO	CARDENAS BARRETO
6	1110176094	IVI DANIELA	CRUZ LEITON
7	1107979724	SHARICK TALIANA	DIAZ CABEZAS
8	1111264796	JESUS JAVIER	GAITA TAPIERO
9	1012373657	MICHELL VALENTINA	GUEVARA PRIETO
10	1111264449	STEFANY	LEYTON DUCUARA
11	1110175828	DAGMARLY	LOZANO MENDEZ
12	1030283140	MARIANA	MACIAS CEDEÑO
13	1111264956	KAREN VALENTINA	MENDEZ REMICIO
14	1110176236	DANEYIS	MONTOYA BETANCOURT
15	1111264214	YERSON	RAMIREZ SANCHEZ
16	1111264612	KEVIN ALEJANDRO	SANCHEZ RODRIGUEZ
17	1111264795	CRISTIAN ANDRES	TOTENA OYUELA
18	1110175324	JEISON JAVIER	URUEÑA TAPIERO

- **Relación de aprendices pendientes por retiro:** No se tienen aprendices para retiro.

**4. Cierre:** Se finaliza la formación.

#### CONCLUSIONES

- Se llamó a lista iniciando la formación y después de terminar el descanso, y todos los aprendices asistieron a la formación y regresaron después del descanso.
- Se socializo la guía de aprendizaje según la competencia programada.
- El aprendiz realizo entrega de las evidencias de desempeño y producto de acuerdo con la lista de chequeo.

#### ESTABLECIMIENTO Y ACEPTACIÓN DE COMPROMISOS

ACTIVIDAD /DECISIÓN	FECHA	RESPONSABLE	FIRMA O PARTICIPACIÓN VIRTUAL


**DE: ASISTENTES Y APROBACIÓN DECISIONES**




NOMBRE	DEPENDENCIA/ EMPRESA	APRUEB A (SI/NO)	OBSERVACIÓN	FIRMA O PARTICIPACIÓN VIRTUAL

De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.

#### ANEXOS

  
 REGISTRO DE ASISTENCIA / DÍA 9 DEL MES DE 10 DEL AÑO 202

OBJETIVO (S) Formación profesional integral en la IE JFK - Ditega: Técnico en programación de software Grado 11

No	NOMBRES Y APELLIDOS	No. DOCUMENTO	PLANTA	CONTRATISTA	OTRO (CUAL?)	DEPENDENCIA/ EMPRESA	CORREO ELECTRÓNICO	TÉLEFONO/EXT.	AUTORIZA GRABACIÓN	FIRMA O PARTICIPACIÓN VIRTUAL
	Daniela Campos	1025507440			Aprende	I.E. J.K	danicampos751@gmail.com	301770490		
	Stefany Leyton	1111254449			Aprende	IE J.K	duccorostefany430@gmail.com	3228852237		Stefany L.
	Nandira Macías	1035283110			Aprende	IE J.K	mariana2009macias@gmail.com			Nandira M.
	Maria Del Carmen B.	1111262177			Aprende	I.E. J.K	mariaadacarmenbarragun354@gmail.com	320246446		Maria B.o
	Karen Menéndez	1111265967			Aprende	I.E. J.K	karenmenendez8@gmail.com	310551656		Karen Menéndez
	Santiago Cordero	1197463649			Aprende	I.E. J.K	cardenasbrocktor@gmail.com	314323988		Santiago Cordero
	Cristian Tahir	1111264295			Aprende	I.E. J.K	cristianandresahir@gmail.com	322669674		Cristian Tahir
	Angie Amos	1030283287			Aprende	IE J.K	tiara@gmail.com	314387595		Angie AT
	Saray Alvarado	11074234			Aprende	IE J.K	sarayal140177@gmail.com	301819135		Saray A.
	Yelson Huira	1110195324			Aprende	IE J.K	tesonhuira@gmail.com	320856779		Yelson UT
	Karen Sánchez	1111264642			Aprende	IE J.K	karenchavez@gmail.com	3163535174		Karen
	Dorelys Monroy	110126206			Aprende	IE J.K	dorelysmonroy@gmail.com	317590188		Dorelys
	Michell Echeverría	1030375741			Aprende	IE J.K	michelguezvara@gmail.com	3143521556		Michell E.
	Shanick Taliana	1110093279			Aprende	IE J.K	sopichdiaz@gmail.com	3012265879		Shanick Taliana
	Yerson Ramirez	1111264214			Aprende	IE J.K	yersonramirez@gmail.com	3208356949		Yerson R.

De acuerdo con La Ley 1581 de 2012, Protección de Datos Personales, el Servicio Nacional de Aprendizaje SENA, se compromete a garantizar la seguridad y protección de los datos personales que se encuentran almacenados en este documento, y les dará el tratamiento correspondiente en cumplimiento de lo establecido legalmente.

GOR-F-085 V02