

	ALCALDÍA DE VILLAVICENCIO	FR-1540-GD01	
	INSTITUCIÓN EDUCATIVA CENTAUROS Aprobación oficial No.0552 del 17 de septiembre del 2002 Nit. 822.002014-4 Código DANE 150001004630	Vigencia: 2020	
	APOYO A LA GESTION ACADEMICA	Documento controlado Página 1 de 8	
Docente: Ana Silvia Mateus Reina		Área: Tecnología	
Grado: 9-1	Sede: La Rosita	Fecha: 19 de abril de 2021	
Estándar: Resuelvo problemas utilizando conocimientos tecnológicos y teniendo en cuenta algunas restricciones y condiciones.			
DBA: N/A			
Nombre del estudiante:			

CRONOGRAMA DE CLASES VIRTUALES POR MEET

TEMAS	FECHA DE CLASE VIRTUAL
1	20 de abril de 2021
2	4 de mayo de 2021
3	18 de mayo de 2021
4	1 de junio de 2021

CRONOGRAMA DE ENTREGA DE ACTIVIDADES

ACTIVIDAD	FECHA MÁXIMA DE ENTREGA
1	27 de abril de 2021
2	13 de mayo de 2021
3	25 de mayo de 2021
4	8 de junio de 2021
Autoevaluación	15 de junio de 2021
Notas finales II periodo	22 de junio de 2021
Finalización del II periodo	25 de junio de 2021

TEMA 1. HISTORIA DE LA PROGRAMACIÓN

Ver los siguientes videos si es posible:

<https://www.youtube.com/watch?v=9hUjhlfs-bw>

De las máquinas de calcular a los computadores programables

Desde las primeras calculadoras mecánicas diseñadas en el siglo XVII hasta los años 40 del siglo pasado se han inventado multitud de máquinas y computadores mecánicos, analógicos o electrónicos que han intentado acelerar y mejorar la precisión de los cálculos.

El punto culminante de todos los enfoques mecánicos para realizar cálculos fue el famoso Motor Analítico diseñado por Charles Babbage en 1840. La diferencia fundamental con todos los artefactos anteriores es que se trataba de una máquina de calcular programable mediante tarjetas perforadas (Babbage se inspiró en el telar de Jacquard en el que el diseño de los dibujos de las telas se podía configurar usando tarjetas perforadas). La máquina estaba diseñada para trabajar en base 10 y se podía conseguir que sus cálculos realizaran saltos condicionales y bucles.

Babbage trabajó durante más de 30 años para intentar construir la máquina. Tenía una enorme complejidad para la época y necesitaba muchísima

financiación. En 1871 murió habiendo podido construir sólo una parte.

La matemática Ada Lovelace tuvo un papel fundamental en la divulgación de la máquina, de su sistema de programación y fue la primera que entendió sus posibilidades más allá del cálculo de fórmulas.

De forma poco habitual para la época, Ada fue educada en el campo de las ciencias y de las matemáticas. A principios de 1840, con veinticinco años, conoció el trabajo de Babbage y colaboró con él, dedicándose durante varios años a conocer y estudiar el diseño y el funcionamiento del motor analítico.

En 1843 publicó el trabajo "Sketch of the analytical engine invented by Charles Babbage" en el que describe el Motor Analítico, añade reflexiones propias sobre el alcance del invento y construye un ejemplo completo, con tablas y diagramas, de cómo hacer que la máquina produzca la secuencia de los números de Bernoulli. Se puede considerar estas tablas y diagramas como el primer programa de un computador.

Antes de que existiera ningún computador real, en 1936, el matemático inglés Alan Turing formalizó la idea abstracta de computador, utilizando un modelo muy sencillo de procesamiento: una máquina abstracta con un scanner que lee y escribe 0s y 1s de una cinta infinita (memoria) y se mueve y los escribe en función de una tabla definida en la máquina (programa). Con esta máquina abstracta (Máquina de Turing) Turing explora la idea de lo computable y lo no computable. ¿Existen problemas no computables para los que no es posible inventar un algoritmo que los resuelva? Turing demuestra que sí y establece con su trabajo los límites de la computación.

En el mismo trabajo Turing define el concepto de máquina universal que es capaz de leer de la cinta un programa cualquiera y simular su comportamiento en otra parte de la cinta. Esta idea tuvo un profundo impacto en el desarrollo de los computadores, porque mostraba que es posible escribir programas que tomen como datos otros programas. Esto abre la puerta a la idea de los programas almacenados en memoria (ya que son otros datos más) y a la creación de compiladores e intérpretes.

En la década de 1940 hubo una explosión de máquinas de computación electrónicas y electromecánicas. Fue una década prodigiosa en la que se desarrollaron tecnologías cada vez más rápidas y resistentes, y se consiguieron enormes avances en la velocidad y precisión de los cálculos.

A mitad de esa década, en 1945, John Von Neumann, que trabajaba en la construcción del ENIAC, introdujo un avance fundamental. Propuso su famosa arquitectura en la que por primera vez se proponen las dos ideas claves de los computadores de propósito general: el programa almacenado en memoria y un conjunto de instrucciones de procesamiento que incluye el direccionamiento indirecto.

Y en 1948, tres años después, se construyó en la universidad de Manchester el primer computador electrónico digital de propósito general que utilizaba esta arquitectura (llamado Baby). Fue diseñado por Max Newman usando la tecnología proporcionada por los ingenieros F.C. Williams y Tom Kilburn. Williams había inventado un dispositivo de memoria electrónico (la válvula de Williams) capaz de sustituir las lentas líneas de retardo de mercurio utilizadas hasta ese momento.

La máquina de Manchester fue el primer computador con un conjunto de instrucciones completo, capaz de realizar saltos, condicionales y direccionamiento indirecto. La primera ejecución de un programa fue el 21 de junio de 1948. En ese año Alan Turing se incorporó a la universidad de Manchester, como director del Laboratorio de Computación. Tres años después, con un diseño ampliado en el que también influyó Turing, una versión mucho mayor de la máquina se convirtió en el primer computador disponible comercialmente, el Ferranti Mark I. El primero se instaló en la universidad de Manchester en febrero de 1951, un mes antes que el UNIVAC I fuera entregado al Departamento de Censo de los EEUU. Se vendieron otras 10 máquinas a Gran Bretaña, Canadá, Holanda e Italia.

El primer programa complejo de Inteligencia Artificial, un jugador de damas escrito por Christopher Strachey, se ejecutó en el verano de 1952 en el Ferranti Mark I en el Laboratorio de Computación de Manchester. Strachey escribió el programa animado por Turing y usando el manual de programación del Ferranti que Turing acababa de escribir. Turing participó también en el desarrollo de otros programas de IA, como un jugador de ajedrez basado en heurísticas.

Los primeros lenguajes de programación

Los primeros computadores electrónicos se programaban directamente usando el conjunto de instrucciones del procesador, en código máquina, código hexadecimal

El primer lenguaje de un nivel algo más elevado que el código máquina es el ensamblador. Comienzan a crearse los primeros programas que procesan lenguajes de programación, aunque se trata de programas muy sencillos, ya que hay una relación casi directa entre la notación en ensamblador y el código hexadecimal que produce el ensamblador.

A finales de la década de los 40 se empiezan a intentar resolver con los primeros computadores los primeros problemas matemáticos distintos de operaciones numéricas: codificación y descodificación, problemas combinatorios como el coloreado del mapa o problemas de ordenación.

Uno de los primeros algoritmos de von Neumann realiza una ordenación de un conjunto de números. Von Neumann lo describe en una carta fechada en 1945. Utiliza el conjunto de instrucciones del EDSAC cuando todavía no se había construido. El programa fue estudiado por Donald Knuth en el artículo Von Neumann's first Computer Program, en donde documenta que había un bug en las primeras instrucciones. Es el primer bug escrito del que se tiene historia. Si Von Neumann hubiera podido ejecutar el programa en el EDSAC se hubiera dado cuenta del error y hubiera sido la primera depuración de un programa.

Primer programa de Von Neumann



(Donald Knuth, "Von Neumann's first Computer Program", Journal of the ACM Computing Surveys (CSUR) Surveys, Volume 2 Issue 4, Dec. 1970, Pages 247-260)

El nacimiento de los computadores comerciales

UNIVAC

El UNIVAC fue el primer computador comercial (1951). Con este computadora aparece por primera vez la figura del programador: manuales, cursos de formación, ofertas de empleo, etc.



Instalaciones comerciales del UNIVAC

IBM 704

El IBM 704 fue el otro gran computador comercial de la década de los 50.

Tuvo una difusión mucho mayor que el UNIVAC: centros gubernamentales, universidades.

Los primeros lenguajes de programación de alto nivel se desarrollan para este computador.



Programando los primeros computadores

El UNIVAC I era una máquina interesante para programar, con su almacenamiento basado en líneas de retardo de mercurio y su propensión a fallar. Los programas se introducían en el computador tecléandolos en cintas magnéticas, una innovación importante en ese tiempo.

El trabajo con el IBM 704 en la Universidad de NY fue una experiencia radicalmente distinta de la del UNIVAC I. Fue construido para ejecutar aplicaciones científicas, y su principal innovación era una memoria de núcleo magnético, reemplazando la memoria de tubos Williams del IBM 701. También tenía una unidad aritmética en punto flotante. La máquina tenía el equivalente a 128 KB de memoria principal, 32 KB de memoria secundaria y cintas magnéticas que podía almacenar 5 MB de datos. Operaba a 0.04 MIPS y costó 3 millones de dólares en 1957.

Tomado de George Sadowsky, My Second Computer was a UNIVAC I

ACTIVIDAD 1. HISTORIA DE LA PROGRAMACIÓN

Desarrollar cada uno de los siguientes puntos con su respectivo enunciado en el cuaderno.
Escribir el título de la actividad.

1. Defina con tus propias palabras que es programación en informática.
2. Desarrolle una línea del tiempo de la historia de la programación, indicando fecha, protagonista y suceso.
3. Realice un resumen de la historia de la programación (mínimo una página)
4. Seleccione 5 personajes de la historia de la programación e investigue los aportes que realizó para el mundo de la tecnología.

Ver los siguientes videos si es posible:
<https://youtu.be/akQtuSrr8ig>

Un algoritmo es un conjunto finito de instrucciones precisas que realizan una tarea, la cual, dado un estado inicial, culminará por arrojar un estado final reconocible.

Son por naturaleza secuenciales, aunque varias técnicas y herramientas aplicables a ellos ofrecen mecanismos de repetición. Son herramientas que tomamos prestadas de las matemáticas para poder dar soluciones computacionales a problemas cotidianos. En la práctica la solución de un problema no está compuesta por uno, sino por muchos algoritmos, cada uno encargado de resolver una porción del problema planteado.

En la vida cotidiana, se emplean algoritmos frecuentemente para resolver problemas. Algunos ejemplos son los manuales de usuario, que muestran algoritmos para usar un aparato, o las instrucciones que recibe un trabajador por parte de su patrón. Algunos ejemplos en matemática son el algoritmo de la división para calcular el cociente de dos números, el algoritmo de Euclides para obtener el máximo común divisor de dos enteros positivos, o el método de Gauss para resolver un sistema lineal de ecuaciones.

CLASIFICACIÓN DE ALGORITMOS

Los algoritmos se pueden clasificar en cuatro tipos:

Algoritmo computacional:

Es un algoritmo que puede ser ejecutado en una computadora.

Ejemplo: Fórmula aplicada para un cálculo de la raíz cuadrada de un valor x .

Algoritmo no computacional:

Es un algoritmo que no requiere de una computadora para ser ejecutado.

Ejemplo: Instalación de un equipo de sonido.

Algoritmo cualitativo:

Un algoritmo es cualitativo cuando en sus pasos o instrucciones no están involucrados cálculos numéricos.

Ejemplos: Las instrucciones para desarrollar una actividad física, encontrar un tesoro.

Algoritmo cuantitativo:

Un algoritmo es cuantitativo cuando en sus pasos o instrucciones involucran cálculos numéricos.

Ejemplo: Solución de una ecuación de segundo grado.

Ejemplo de un algoritmo cualitativo:

Algoritmo para hacer un huevo frito

1. Sacar el sartén de la alacena
2. Sacar el huevo de la nevera
3. Sacar el aceite de la alacena
4. Sacar la sal
5. Tomar el fósforo
6. Prender el fosforo
7. Encender la estufa
8. Apagar el fosforo
9. Tomar el sartén
10. Colocar el sartén en la estufa

TEMA 2. ALGORITMOS

11. Dejar calentar un poco el sartén
12. Tomar el aceite
13. Abrir el aceite
14. Echar una cucharadita de aceite al sartén...
15.Continuemos... no hemos terminado quiero desayunar....

ACTIVIDAD 2. ALGORITMOS

A. Desarrollar los siguientes puntos en el cuaderno, indicando el título de la actividad y cada uno de los enunciados

1. Desarrollar y terminar todo el ejemplo de “cómo hacer un huevo frito”
2. Desarrollar los algoritmos detallados paso a paso de las siguientes situaciones:
3. Rutina (algoritmo) que tienes desde que te despiertas hasta que terminas el desayuno.
4. Algoritmo que debes seguir al lavar la ropa.
5. Algoritmo que sigues para hacer el almuerzo.

Puede contar con la ayuda de sus padres.

Recomendación: Enumerar cada paso del algoritmo. Entre más detallados mejor.

B. Desarrollar el ANEXO 1 (Se encuentra al final de la guía). Se debe entregar un video (ver explicación en el ANEXO 1) y el desarrollo de la guía de trabajo del ANEXO 1.

TEMA 3. DIAGRAMAS DE FLUJO

Ver los siguientes videos si es posible:

<https://youtu.be/tMEscFCEP0g>

¿Qué es un diagrama de flujo?

Un diagrama de flujo es un diagrama que describe un proceso, sistema o algoritmo informático. Se usan ampliamente en numerosos campos para documentar, estudiar, planificar, mejorar y comunicar procesos que suelen ser complejos en diagramas claros y fáciles de comprender.

Los diagramas de flujo emplean rectángulos, óvalos, diamantes y otras numerosas figuras para definir el tipo de paso, junto con flechas conectoras que establecen el flujo y la secuencia. Pueden variar desde diagramas simples y dibujados a mano hasta diagramas exhaustivos creados por computadora que describen múltiples pasos y rutas.

Reglas Básicas Para la Construcción de un Diagrama de Flujo

1. Todos los símbolos han de estar conectados
2. A un símbolo de proceso pueden llegarle varias líneas
3. A un símbolo de decisión pueden llegarle varias líneas, pero sólo saldrán dos (Si o No, Verdadero o Falso).
4. A un símbolo de inicio nunca le llegan líneas.
5. De un símbolo de fin no parte ninguna línea.
6. Debe tener un fin en algún lugar del diagrama.

Los símbolos que se usan para realizar los diagramas de flujo son los siguientes:

Símbolo	Nombre	Función
	Inicio / Final	Representa el inicio y el final de un proceso
	Línea de Flujo	Indica el orden de la ejecución de las operaciones. La flecha indica la siguiente instrucción.
	Entrada / Salida	Representa la lectura de datos en la entrada y la impresión de datos en la salida
	Proceso	Representa cualquier tipo de operación
	Decisión	Nos permite analizar una situación, con base en los valores verdadero y falso

Ejemplo

Diagrama 1

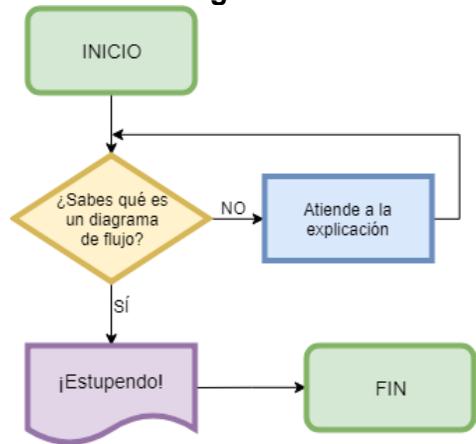


Diagrama 2

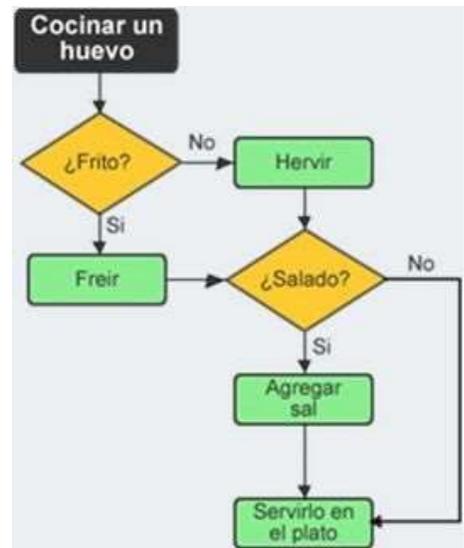
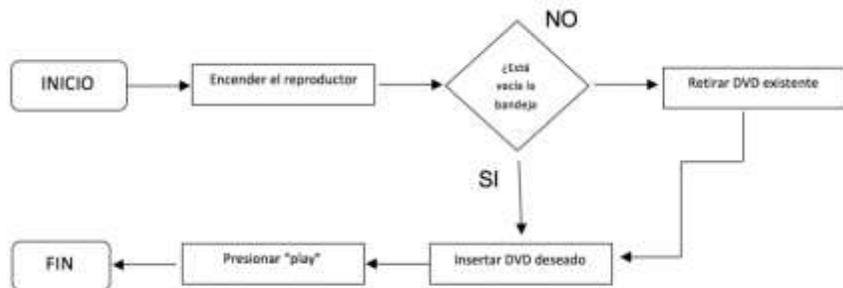


Diagrama 3



ACTIVIDAD 3. DIAGRAMA DE FLUJO

Desarrollar la siguiente actividad en el cuaderno, no olvides copiar los enunciados también, tener buena presentación en los diagramas de flujo teniendo en cuenta las reglas.

1. Asigne un nombre que represente la funcionalidad del diagrama de flujo llamado Diagrama 1.
2. Asigne un nombre que represente la funcionalidad del diagrama de flujo llamado Diagrama 2.
3. Asigne un nombre que represente la funcionalidad del diagrama de flujo llamado Diagrama 3.
4. ¿Qué diferencia observa en el diagrama de flujo 1, 2 y 3? Ten en cuenta las reglas básicas de la construcción de los diagramas de flujo
5. Realizar los siguientes diagramas de flujo.
 - a. Pedir un domicilio
 - b. Suma de dos números
 - c. Ir a comprar ropa en un almacén
 - d. Suma de 4 números.
 - e. Calcule el perímetro de un cuadrado.
 - f. Calcule el área de un triángulo.

TEMA 4. LENGUAJES DE PROGRAMACIÓN

Ver los siguientes videos si es posible:

<https://www.youtube.com/watch?v=pWw4UtQhdek>

Un programa es un conjunto de órdenes empleado para que el computador realice una determinada tarea.

Existen muchos conjuntos de instrucciones que sirven para crear un programa. A cada uno de estos conjuntos se le llama **lenguaje de programación**.

Alguno de los lenguajes de programación más utilizados son JAVA, C++, Visual Basic, Python, JavaScript, MathLab...

Existen algunos lenguajes de programación específicos que tienen como finalidad enseñar a programar a principiantes. Algunos de estos lenguajes son Scratch, Processing, Alice...

Existen dos tipos de lenguaje de programación:

1. Lenguajes textuales:

Son aquellos en los que las órdenes que se insertan en el programa se escriben en forma de texto. El conjunto de todas las instrucciones de un programa se denomina código. Normalmente estos lenguajes se han desarrollado en inglés, por lo que las instrucciones coinciden con palabras inglesas. Por ejemplo: Repeat, una instrucción que sirve para repetir órdenes. O IF, IF ELSE, que sirven para tomar una decisión en función de una condición. WHILE, que sirve para ejecutar una parte del código escrito mientras se está cumpliendo cierta condición, etc.

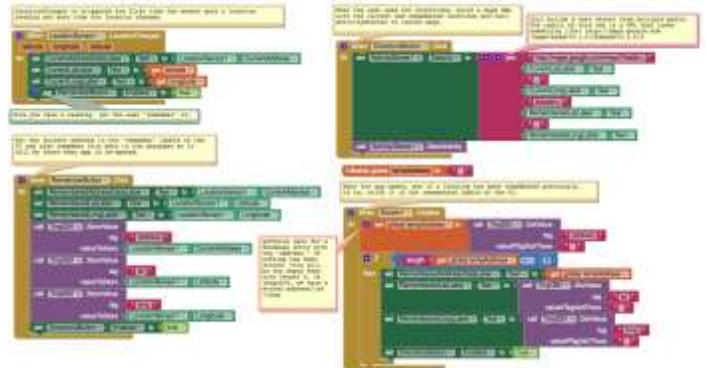
Estos lenguajes textuales son los más potentes y usados por profesionales, ya que se obtienen programas rápidos y versátiles. Hay ciertas cosas que un programa puede hacer gracias a que se ha escrito el código de forma correcta y que no se

podrían hacer de forma tan sencilla utilizando un lenguaje gráfico.

```
1 // class declaration
2 public class ProgrammingExample {
3
4 // method declaration
5 public void sayHello() {
6
7 // method output
8 System.out.println("Hello World!");
9 }
10 }
```

2. Lenguajes gráficos.

Las instrucciones vienen dadas por bloques gráficos predefinidos. Es muy intuitivo y se basa en encajar estos bloques gráficos para realizar diferentes acciones.



Lenguaje de programación es el "idioma" utilizado para controlar el comportamiento del computador o cualquier máquina controlada por computador. Consiste en un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

En la actualidad los lenguajes de programación están estructurados y escritos para ser comprensibles por el ser humano. A este código se le llama **código fuente** que no es comprendido todavía por el computador, ya que este sólo maneja el lenguaje binario de 0 y 1, es decir, si hay corriente o no hay corriente.

La **compilación** es el proceso de traducir un programa escrito en código fuente a un programa en código objeto que es el que utiliza el lenguaje binario. El programa encargado de compilar es compilador. La mayoría del software de programación trae su propio compilador.

Clasificación de lenguajes de programación

Los lenguajes de programación se clasifican en varios tipos colocándolos en diferentes niveles. Hay lenguajes de programación de alto nivel y lenguajes de programación de bajo nivel.

- ✓ *Lenguaje de programación de alto nivel*
Los lenguajes de alto nivel son normalmente fáciles de aprender porque están formados por elementos de lenguajes naturales, como el inglés.

Ejemplos de lenguajes de alto nivel: BASIC, Python, Pascal, MATLAB...

Ejemplos de lenguajes de bajo nivel: Lenguaje máquina...

✓ *Lenguajes de bajo nivel*

Son lenguajes de programación que se acercan al funcionamiento de una computadora.

El lenguaje de más bajo nivel es, por excelencia, el código máquina. A éste le sigue el lenguaje ensamblador, ya que al programar en ensamblador se trabajan con los registros de memoria de la computadora de forma directa.

ACTIVIDAD 4. LENGUAJES DE PROGRAMACION

Desarrollar los siguientes puntos en el cuaderno.

1. Realizar un mapa conceptual sobre los lenguajes de programación utilizando la información que esta en esta guía.
2. Realice una consulta sobre los 3 lenguajes de programación actuales y cuáles son las características de cada uno.

ANEXO 1. PREPARACION RECETA DE COCINA

Proyecto Integración

Materia: Tecnología e Informática

Tema: Algoritmos

Docente: Ana Silvia Mateus Reina

Descripción:

Con este proyecto el estudiante creará UN VIDEO mostrando la preparación de una receta de cocina; en donde se debe mostrar los ingredientes requeridos para un número determinado de porciones; la preparación de dicha receta paso a paso, además de la forma de servir el plato.

Objetivos:

*El estudiante analizará los procesos para preparar una receta de cocina.

*Calcular la cantidad de los ingredientes teniendo presente la cantidad de porciones que se requieren.

Requisitos:

- Contar con la presencia de un adulto (padre de familia y/o acudiente) en todo el proceso.
 - Seleccionar receta de cocina dependiendo de sus intereses.
 - Contar con las normas de seguridad que les permita realizar un trabajo de manera segura y tranquila.
-

Recursos:

*Ingredientes requeridos para la preparación de la receta.

*Elementos requeridos para la preparación de la receta.

Labor del Docente:

1. Entrega formato de la guía para el desarrollo del trabajo.
2. Calificación del trabajo.

Labor del Estudiante:

1. Elección de la receta de cocina.
 2. Desarrollar la guía de trabajo.
-

Criterios de Evaluación:

- ✓ Acompañamiento adulto (padre de familia y/o acudiente)
- ✓ Selección receta.
- ✓ Calculo de cantidad de ingredientes Vs cantidad de personas.
- ✓ Relación de gastos (Presentar copia factura)
- ✓ Escribir en la guía el paso a paso de la receta en la parte de PREPARACIÓN.

Entregables:

1. Guía de trabajo diligenciada – SI NO ALCANZA LOS ESPACIOS DEL CUADRO PUEDEN UTILIZAR LA PARTE DE ATRÁS.
2. Relación de gastos



GUIA DE TRABAJO



Nombre del Estudiante		
Adulto acompañante:	Nombre:	Firma:
Nombre receta:		
Cantidad porciones:		
Tiempo Preparación:		
Ingredientes:		
<p> Preparación (Enumere paso a paso el proceso para la preparación) Puede utilizar la parte de atrás de la hoja </p>		
Presupuesto de la receta Ingredientes – valor (Puede utilizar la parte de atrás de la hoja)		