

INSTITUCIÓN EDUCATIVA FE Y ALEGRÍA AURES

GUIA DIDÁCTICA SEGUNDO PERIODO – TECNOLOGÍA E INFORMÁTICA

IDENTIFICACIÓN

DOCENTE	MARÍA REINALDA CASTAÑO Correo: reinacas@hotmail.com Celular: 3043650640 Enlace Virtual: https://meet.google.com/dzn-ydve-ift			GRADO	DÉCIMO	
TIPO DE GUIA:	REPASO	X	INFORMATIVA	X	EJERCITACIÓN	X
DURACIÓN	10 SEMANAS INICIO ABRIL 5 DE 2021 INTENSIDAD SEMANAL 2H					
INDICADORES DE DESEMPEÑO	Utiliza herramientas de colaboración digital para compartir documentos, experiencias y conocimientos. Soluciona una situación planteada, apoyándose en conceptos y estructuras propias de la programación. Conoce los símbolos que se utilizan para representar algoritmos mediante diagramas de flujo. Analiza problemas, utilizando una metodología de pasos ordenados para darles solución; teniendo en cuenta las necesidades económicas, sociales y culturales de su entorno.					
CONTENIDO	ALGORITMOS					

EVALUACIÓN.

Cada actividad hará parte de 40% del seguimiento de la asignatura, el examen de periodo equivale al 40% de la asignatura y el 20% evalúa la actitud que asume el estudiante frente al desarrollo de la asignatura **TECNOLOGÍA E INFORMÁTICA.**

BIBLIOGRAFÍA:

Se otorga permiso para enlazar este documento desde cualquier sitio Web, con la siguiente dirección: <http://www.eduteka.org/GuiaAlgoritmos.php>

MEDIOS Y FECHA DE ENTREGA

Enviar a la docente María Reinalda Castaño Correo: reinacas@hotmail.com, Celular: 3043650640

Cuaderno enviado al colegio, o llamar al teléfono fijo 5834728 y dar respuesta oral a las actividades propuestas.

No importa que medio utilice para hacer entrega de las actividades siempre deben contener: nombres y apellidos del estudiante, el grado o grupo, la asignatura, número de la actividad que se está presentando al docente, el nombre del docente y enumerar las páginas antes de enviarlas.

Fecha de entrega: se reciben actividades hasta la octava semana del periodo, la novena y décima semana es para realizar el examen de periodo y la evaluación actitudinal.

ALGORITMOS, CONCEPTOS BÁSICOS

1. ¿Qué es un algoritmo?
2. Representación de algoritmos
3. Simbología de los diagramas de flujo
4. Reglas para la elaboración de diagramas de flujo
5. Conceptos básicos de programación
6. Variables
7. Constantes
8. Contadores
9. Acumuladores
10. Identificadores
11. Palabras reservadas
12. Funciones matemáticas
13. Tipos de datos
14. Operadores
15. Orden de evaluación de los operadores
16. Expresiones

¿Qué es un algoritmo?

En el ámbito de la computación, **los Algoritmos** son una herramienta que permite describir claramente un conjunto finito de instrucciones, ordenadas secuencialmente y libres de ambigüedad, que debe llevar a cabo un computador para lograr un resultado previsible. Vale la pena recordar que un programa de computador consiste de una serie de instrucciones muy precisas y escritas en un lenguaje de programación que el computador entiende (Logo, Java, Pascal, etc.) .

En resumen, un **Algoritmo** es una secuencia ordenada de instrucciones, pasos o procesos que llevan a la solución de un determinado problema. Los hay tan sencillos y cotidianos como seguir la receta del médico, abrir una puerta, lavarse las manos, hasta los que conducen a la solución de problemas muy complejos.

EJEMPLO

Un procedimiento que realizamos varias veces al día consiste en lavarnos los dientes. Veamos la forma de expresar este procedimiento como un Algoritmo:

1. Tomar la crema dental
2. Destapar la crema dental
3. Tomar el cepillo de dientes
4. Aplicar crema dental al cepillo
5. Tapar la crema dental
6. Abrir la llave del lavamanos
7. Remojar el cepillo con la crema dental
8. Cerrar la llave del lavamanos
9. Frotar los dientes con el cepillo
10. Abrir la llave del lavamanos
11. Enjuagarse la boca
12. Enjuagar el cepillo

13. Cerrar la llave del lavamanos
14. Secarse la cara y las manos con una toalla

EJEMPLO

El ejemplo de cambiar una bombilla (foco) fundida es uno de los más utilizados por su sencillez para mostrar los pasos de un Algoritmo:

1. Ubicar una escalera debajo de la bombilla fundida
2. Tomar una bombilla nueva
3. Subir por la escalera
4. Girar la bombilla fundida hacia la izquierda hasta soltarla
5. Enroscar la bombilla nueva en el plafón hasta apretarla
6. Bajar de la escalera
7. Fin

En términos generales, un **Algoritmo** debe ser:

- Realizable:** El proceso algorítmico debe terminar después de una cantidad finita de pasos. Se dice que un algoritmo es inaplicable cuando se ejecuta con un conjunto de datos iniciales y el proceso resulta infinito o durante la ejecución se encuentra con un obstáculo insuperable sin arrojar un resultado.
- Comprensible:** Debe ser claro lo que hace, de forma que quien ejecute los pasos (ser humano o máquina) sepa qué, cómo y cuándo hacerlo. Debe existir un procedimiento que determine el proceso de ejecución.
- Preciso:** El orden de ejecución de las instrucciones debe estar perfectamente indicado. Cuando se ejecuta varias veces, con los mismos datos iniciales, el resultado debe ser el mismo siempre. La precisión implica determinismo.

Los algoritmos no admiten ningún tipo de ambigüedad ya que los lenguajes de programación tienen un vocabulario restringido y preciso. Esto exige la utilización de un conjunto determinado de palabras, mandos o primitivas en cualquiera de los procedimientos que se elaboren.

REPRESENTACIÓN DE ALGORITMOS

Los Algoritmos se puede expresar de muchas maneras, pero en esta guía se tratarán solo dos formas: Seudocódigo y Diagrama de Flujo. En **Seudocódigo** la secuencia de instrucciones se representa por medio de frases o proposiciones, mientras que en un **Diagrama de Flujo** se representa por medio de gráficos.

Elseudocódigo está compuesto por proposiciones informales en español que permiten expresar detalladamente las instrucciones que llevan desde un estado inicial (problema) hasta un resultado deseado (solución). Por lo regular, los algoritmos se escriben por refinamiento: se escribe una primera versión que luego se descompone en varios subproblemas (el número depende de la complejidad del problema) independientes entre sí. Si es necesario se va refinando cada vez las instrucciones hasta que las proposiciones generales en español como las del ejemplo anterior se puedan codificar en el lenguaje seleccionado para hacer la programación.

El **Diagramas de Flujo** para representar un algoritmo tiene claras ventajas, especialmente cuando son construidos por estudiantes de básica y media. Numerosas investigaciones han mostrado que el Aprendizaje Visual es uno de los mejores métodos para enseñar habilidades del pensamiento. Las técnicas que utilizan formas graficas para representar ideas e información ayudan a los estudiantes a clarificar su pensamiento, y a procesar, organizar y priorizar nueva información. Los diagramas visuales revelan patrones, interrelaciones e interdependencias además de estimular el pensamiento creativo.

La utilización de Diagramas ayuda a los estudiantes a:

- *Clarificar el pensamiento:* Ellos pueden ver cómo se conectan los procesos y se dan cuenta de cómo estos se pueden organizar o agrupar para darles el orden lógico correcto.
- *Identificar pasos erróneos:* Sobre un diagrama es más fácil identificar los cambios que se requieren para el correcto funcionamiento de un programa de computador que hacerlo sobre el código.

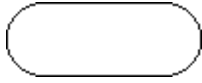
Los Diagramas de Flujo son una de las técnicas más utilizadas para representar gráficamente la secuencia de instrucciones de un Algoritmo. Estas instrucciones están compuestas por operaciones, decisiones lógicas y ciclos repetitivos, entre otros. La solución de un problema puede contener varios conjuntos de instrucciones (procedimientos o métodos) que tienen como finalidad ejecutar cada uno de los procesos necesarios para llegar a la solución de un problema a partir de los datos disponibles (estado inicial).

Las ventajas de diseñar un **Diagrama de Flujo** antes de empezar a generar el código de un programa son:

- Forzar la identificación de todos los pasos de una solución de forma clara y lógica;
- Establecer una visión amplia y objetiva de la solución;
- Verificar si se han tenido en cuenta todas las posibilidades;
- Comprobar si hay procedimientos duplicados;
- Representar gráficamente una solución (es más simple hacerlo con gráficas que mediante palabras);
- Facilitar a otras personas la comprensión de la secuencia lógica de la solución planteada;
- Posibilitar acuerdos con base en la aproximación común a una solución de un problema, resolver ambigüedades o realizar mejoras;
- Establecer posibles modificaciones (resulta más fácil depurar un programa con el diagrama que con el listado del código);
- Agilizar la codificación (traducción) del algoritmo en un lenguaje de programación;
- Servir como elemento de documentación de la solución del problema.

SIMBOLOGÍA DE LOS DIAGRAMAS DE FLUJO

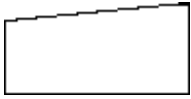
La estandarización de los símbolos para la elaboración de Diagramas de Flujo tardó varios años. Con el fin de evitar la utilización de símbolos diferentes para representar procesos iguales, la Organización Internacional para la Estandarización (ISO, por su sigla en inglés) y el Instituto Nacional Americano de Estandarización (ANSI, por su sigla en inglés), estandarizaron los símbolos que mayor aceptación tenían en 1985. Los siguientes son los principales símbolos para elaborar Diagramas de Flujo:

**Inicio/Final**

Se utiliza para indicar el inicio y el final de un diagrama; del Inicio sólo puede salir una línea de flujo y al Final sólo debe llegar una línea.

**Entrada General**

Entrada/Salida de datos en General (en esta guía, solo la usaremos para la Entrada).

**Entrada por teclado**

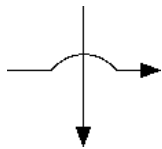
Instrucción de entrada de datos por teclado. Indica que el computador debe esperar a que el usuario teclee un dato que se guardará en una variable o constante.

**Llamada a subrutina**

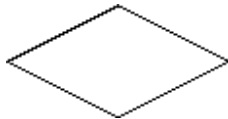
Indica la llamada a una subrutina o procedimiento determinado.

**Acción/Proceso General**

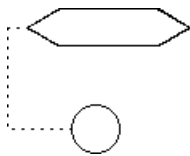
Indica una acción o instrucción general que debe realizar el computador (cambios de valores de variables, asignaciones, operaciones aritméticas, etc.).

**Flujo**

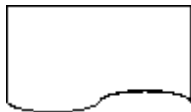
Indica el seguimiento lógico del diagrama. También indica el sentido de ejecución de las operaciones.

**Decisión**

Indica la comparación de dos datos y dependiendo del resultado lógico (falso o verdadero) se toma la decisión de seguir un camino del diagrama u otro.

**Iteración**

Indica que una instrucción o grupo de instrucciones deben ejecutarse varias veces.

**Salida Impresa**

Indica la presentación de uno o varios resultados en forma impresa.

**Salida en Pantalla**

Instrucción de presentación de mensajes o resultados en pantalla.

**Conector**

Indica el enlace de dos partes de un diagrama dentro de la misma página.

**Conector**

Indica el enlace de dos partes de un diagrama en páginas diferentes.

El Diagrama de Flujo es una herramienta gráfica valiosa para la representación esquemática de la secuencia de instrucciones de un **algoritmo** o de los pasos de un proceso.

EJEMPLO

Elaborar un Algoritmo para calcular el área de cualquier triángulo rectángulo y presentar el resultado en pantalla.

SEUDOCÓDIGO

Paso 1: Inicio

Paso 2: Asignar el número 2 a la constante "Div"

Paso 3: Conocer la base del triángulo y guardarla en la variable "Base"

Paso 4: Conocer la altura del triángulo y guardarla en la variable "Altura"

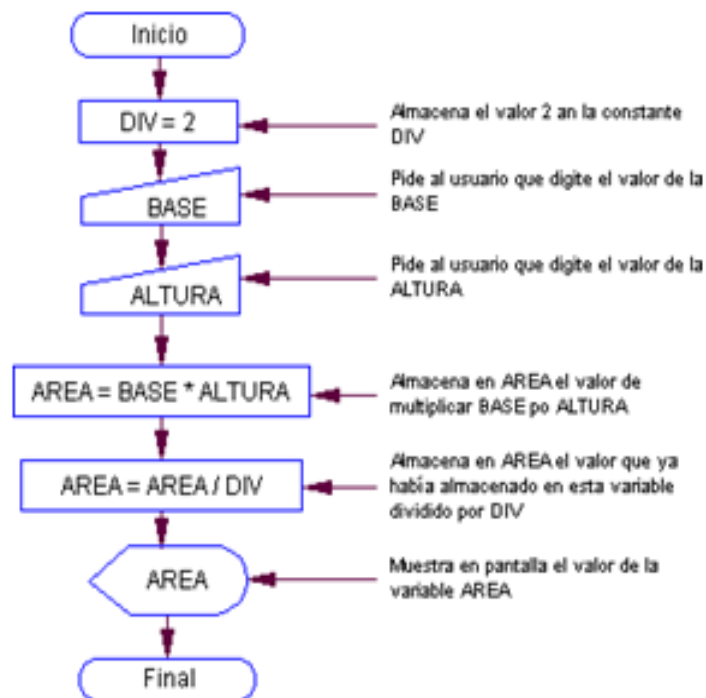
Paso 5: Guardar en la variable "Area" el valor de multiplicar "Base" por "Altura"

Paso 6: Guardar en la variable "Area" el valor de dividir "Area" entre "Div"

Paso 7: Reportar el valor de la variable "Area"

Paso 8: Final

DIAGRAMA DE FLUJO



REGLAS PARA LA ELABORACIÓN DE DIAGRAMAS DE FLUJO

Cuando el algoritmo se desea expresar en forma de diagrama de flujo, se deben tener en cuenta algunas reglas o principios básicos para su elaboración (Rojas & Nacato, 1980):

- Poner un encabezado que incluya un título que identifique la función del algoritmo; el nombre del autor; y la fecha de elaboración;
- Sólo se pueden utilizar símbolos estándar (ISO 5807);
- Los diagramas se deben dibujar de arriba hacia abajo y de izquierda a derecha;
- La ejecución del programa siempre empieza en la parte superior del diagrama;
- Los símbolos de "Inicio" y "Final" deben aparecer solo una vez;
- La dirección del flujo se debe representar por medio de flechas (líneas de flujo);
- Todas las líneas de flujo deben llegar a un símbolo o a otra línea;
- Una línea de flujo recta nunca debe cruzar a otra. Cuando dos líneas de flujo se crucen, una de ellas debe incluir una línea arqueada en el sitio donde cruza a la otra (ilustración 2-5);
- Se deben inicializar las variables que se utilicen o permitir la asignación de valores mediante consulta al usuario;
- Las bifurcaciones y ciclos se deben dibujar procurando una cierta simetría;
- Cada rombo de decisión debe tener al menos dos líneas de salida (una para SI y otra para NO);
- Las acciones y decisiones se deben describir utilizando el menor número de palabras posible; sin que resulten confusas o poco claras;
- Si el Diagrama se vuelve complejo y confuso, es mejor utilizar símbolos conectores para reducir las líneas de flujo;
- Todo el Diagrama debe ser claro, ordenado y fácil de recorrer;
- El Diagrama se debe probar recorriéndolo con datos iniciales simples (prueba de escritorio).

EJEMPLO DE ALGORITMO REPRESENTADO EN SEUDOCÓDIGO Y EN DIAGRAMA DE FLUJO

EJEMPLO

Elabore el algoritmo de un contador de 15 personas o estudiantes que pueden ingresar al aula de clase en modalidad presencial, una vez hayan ingresado estos estudiantes la puerta del aula se cierra para evitar el ingreso de más personas o estudiantes, el ingreso al aula es de a uno para impedir congestión en la puerta. Muestre en una pantalla que ya el grupo de 15 personas esta completo.

Realizar el algoritmo en pseudocódigo y en diagrama de flujo.

SOLUCIÓN

Seudocódigo

Paso 1: Inicio

Paso 2: abrir la puerta

Paso 3: Asignar el número 0 a la variable "CONTADOR"

Paso 4: mostrar el valor registrado en la variable "CONTADOR" EN PANTALLA

Paso 5: permitir el ingreso a una persona

Paso 6: A la variable $CONTADOR = CONTADOR + 1$

Paso 7: preguntar la variable $CONTADOR$ ¿es igual a 15?

Paso 8: si la variable $CONTADOR$ no es igual a 15 se repiten los pasos 4,5,6 y 7

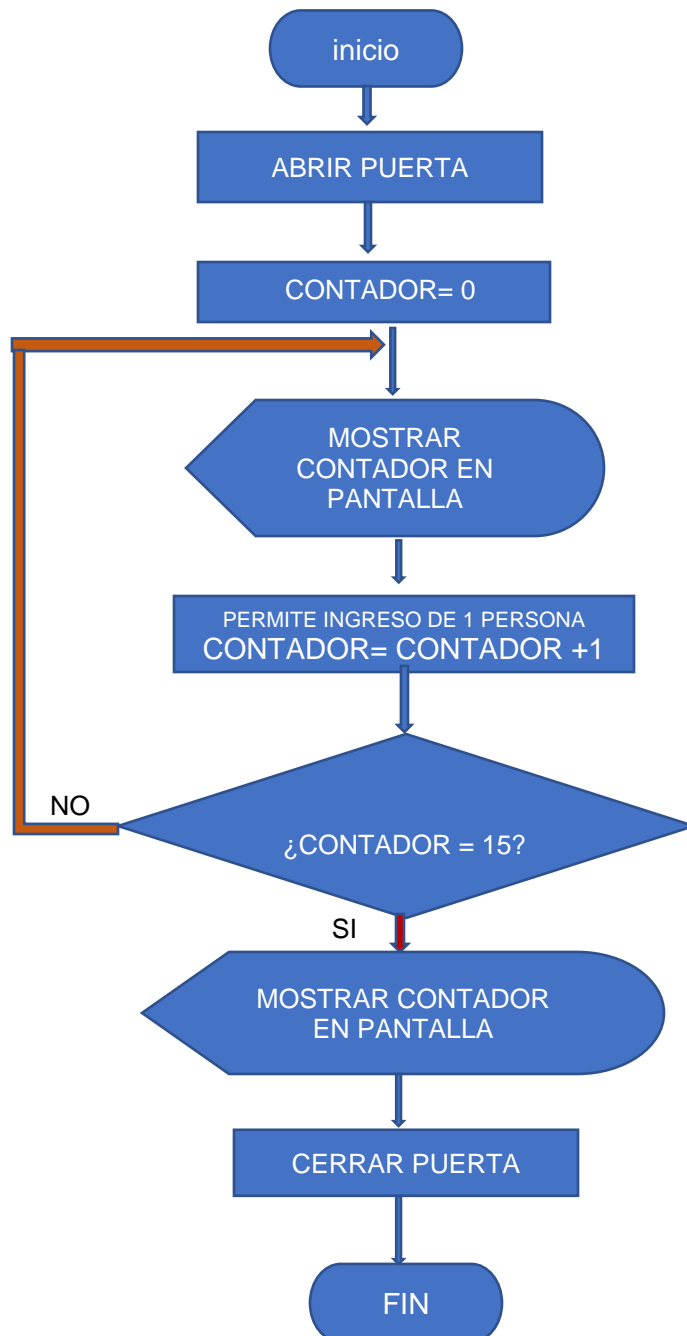
Paso 9: si la variable $CONTADOR$ es igual a 15 se realiza el paso 4

Paso 10: Se cierra la puerta

Paso 11: Si la puerta se abre el contador se repite todo el proceso desde el paso 3

Paso 10: Final.

DIAGRAMA DE FLUJO

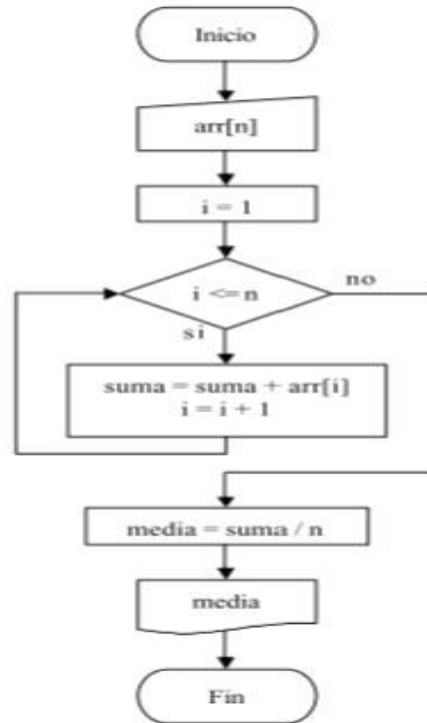


Ejemplo

Elabore el algoritmo para obtener la media de n números

- Pseudocódigo**
1. Obtener un arreglo de n números
 2. Desde $i=1$ hasta n
 3. Hacer $suma = suma + arreglo[i]$
 4. Hacer $media = suma / n$
 5. Imprimir $media$

Diagrama de flujo



EJEMPLO

Realizar el algoritmo que convierte grados Fahrenheit a grados Celsius

Dato de entrada:

F: los grados Fahrenheit a convertir
Numérico

Dato de salida:

C: los grados Celsius resultantes
Numérico

Fórmula:

$$C = \frac{5}{9}(F - 32)$$

Pseudocódigo y diagrama de flujo

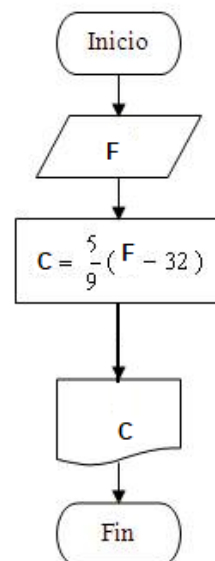
Inicio

Leer F

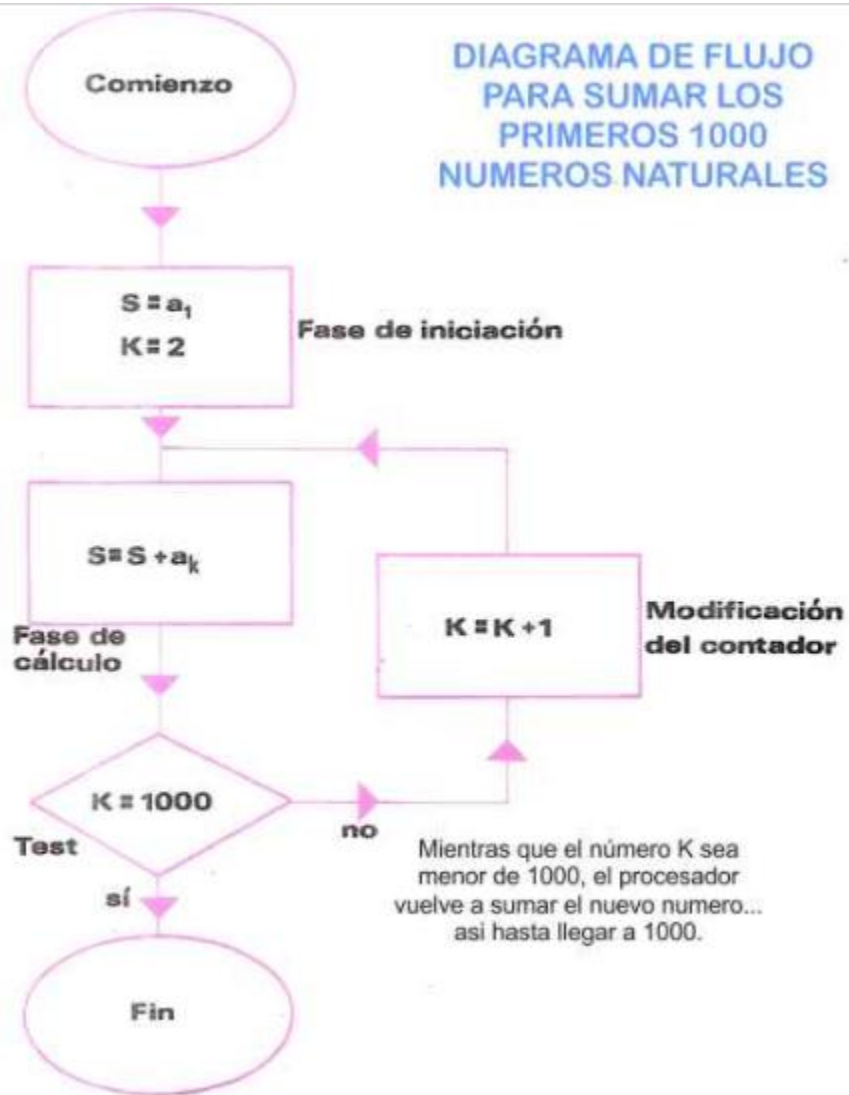
$$C = \frac{5}{9}(F - 32)$$

Escribir C

Fin algoritmo



EJEMPLO



Significa que el proceso comienza con un valor para el contador de $k = 2$, de tal forma que la primera operación elemental realiza la suma $a_1 + a_2$.

Como el valor del contador no es 1.000 como indica el test, el valor de k cambia a $k + 1$, es decir, $2+1=3$. siendo la próxima operación la suma anterior más a_3 .

Este proceso continúa hasta que k alcanza el valor de 1.000, momento en el cual el **algoritmo** termina. Es importante señalar que en informática el signo $=$ no tiene el mismo sentido que en matemáticas (donde, evidentemente, la expresión $k = k + 1$ es falsa), sino un sentido de «sustituir por».

ACTIVIDADES Y EXAMEN SOBRE ALGORITMOS

ACTIVIDAD #1

Realizar un algoritmo en pseudocódigo y diagrama de flujo, sobre el proceso de matrícula de estudiantes nuevos en la Institución educativa Fe y Alegría Aures.

ACTIVIDAD #2

Realizar un algoritmo en pseudocódigo y diagrama de flujo, sobre el proceso de evaluación y el resultado de un periodo académico según las indicaciones del SIEPE de la Institución educativa Fe y Alegría Aures. Mostrar la nota final de periodo en pantalla.

Recuerde, se evalúa con nota numérica de 1 a 5, 40% actividades, talleres, tareas; 40% examen de periodo y 20% actitudinal (6% heteroevaluación, 7% coevaluación y 7% autoevaluación).

EXAMEN DEL SEGUNDO PERIODO

INSTITUCIÓN EDUCATIVA FE Y ALEGRÍA AURES

EXAMEN DE TECNOLOGÍA E INFORMÁTICA TEMA: ALGORITMOS

GRUPO: DECIMO SEGUNDO PERIODO AÑO: 2021

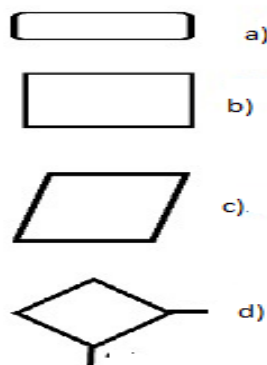
NOMBRE COMPLETO DEL ESTUDIANTE: _____

PROFESORA: MARÍA REINALDA CASTAÑO

- serie ordenada de pasos que sirven para solucionar un problema o tarea
 - Diagrama de flujo
 - Algoritmo
 - Lenguaje de programación
- un algoritmo que nos asegura que si seguimos más de una vez la serie de pasos Descritos llegaremos al mismo resultado es:
 - Definido
 - Finito
 - preciso
- Utiliza símbolos para identificar las instrucciones a realizar dentro de un programa.
 - algoritmo
 - proceso
 - diagrama de flujo
- un algoritmo que cuenta con un determinado número de pasos, indicando un inicio y un fin es:
 - Finito
 - Definido
 - Preciso
- se utiliza para indicar la dirección que lleva el diagrama de flujo.
 - Círculo
 - flecha
 - cuadrado
- el algoritmo al indicar el orden en que debe realizarse cada paso es:
 - Finito
 - Preciso
 - definido
- Son dos características de los algoritmos.

- a) Corto y rápido
 - b) Finito y preciso
 - c) Rápido y correcto
8. el primer paso para solucionar un problema por medio de UN ALGORITMO es:
- a) análisis de la solución
 - b) definición del problema
 - c) diseño de la solución
 - d) codificación
9. Es la representación gráfica de un algoritmo.
- a) Símbolos
 - b) Anuncios
 - c) Diagrama de flujo
10. teniendo en cuenta que Un algoritmo debe definir tres partes: **Entrada, Proceso y Salida**.
En el siguiente ejemplo: el algoritmo de una receta de cocina, seleccionar la ENTRADA
- a) ingrediente y utensilios.
 - b) elaboración de la receta en la cocina.
 - c) terminación del plato (por ejemplo, Pollo al horno)
11. teniendo en cuenta que Un algoritmo debe definir tres partes: **Entrada, Proceso y Salida**.
En el siguiente ejemplo: el algoritmo de una receta de cocina, seleccionar la SALIDA
- a) ingrediente y utensilios.
 - b) elaboración de la receta en la cocina.
 - c) terminación del plato (por ejemplo, Pollo al horno)
12. teniendo en cuenta que Un algoritmo debe definir tres partes: **Entrada, Proceso y Salida**.
En el siguiente ejemplo: el algoritmo de una receta de cocina, seleccionar el PROCESO
- a) ingrediente y utensilios
 - b) elaboración de la receta en la cocina.
 - c) terminación del plato (por ejemplo, Pollo al horno)

Para las preguntas del 13 al 16 utilice la siguiente imagen



13. el símbolo a) se utiliza en un diagrama de flujo para representar

- a. un proceso
- b. una entrada y/o salida de datos
- c. toma de decisión
- d. el inicio o el fin de un proceso

14. el símbolo b) se utiliza en un diagrama de flujo para representar

- a. un proceso
- b. una entrada y/o salida de datos
- c. toma de decisión
- d. el inicio o el fin de un proceso

15. el símbolo c) se utiliza en un diagrama de flujo para representar

- a. un proceso
- b. una entrada y/o salida de datos
- c. toma de decisión
- d. el inicio o el fin de un proceso

16. el símbolo d) se utiliza en un diagrama de flujo para representar

- a. un proceso
- b. una entrada y/o salida de datos
- c. toma de decisión
- d. el inicio o el fin de un proceso

17. el siguiente símbolo se utiliza en un diagrama de flujo para representar:



- a. una pantalla
- b. una impresora
- c. un teclado
- d. un archivo en cinta magnética

18. el siguiente símbolo se utiliza en un diagrama de flujo para representar:



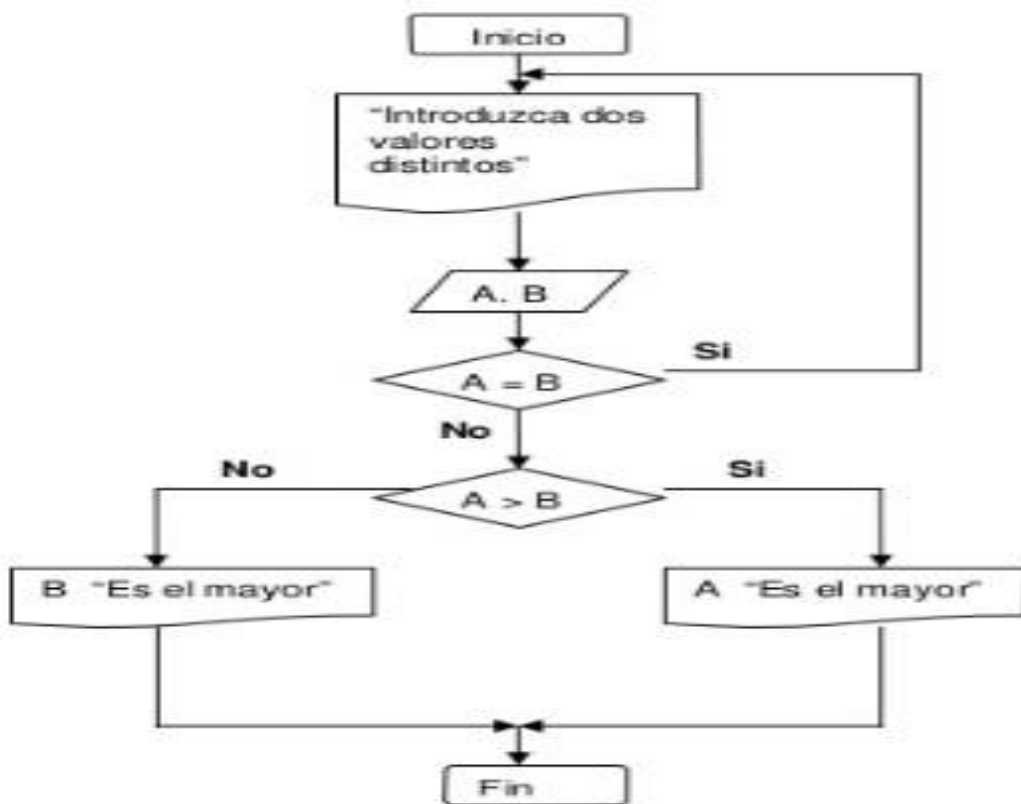
- a. una pantalla
- b. una impresora
- c. un teclado
- d. un archivo en cinta magnética

19. el siguiente símbolo se utiliza en un diagrama de flujo para representar:



- a. una pantalla
- b. una impresora
- c. un teclado
- d. un archivo en cinta magnética

20. el siguiente diagrama de flujo permite:



www.areatecnologia.com

- a. leer dos números diferentes y mostrar el menor de los dos números
- b. leer dos números diferentes y mostrar el mayor y el menor de los dos números
- c. leer dos números diferentes y mostrar cuando los dos números son iguales.
- d. Leer dos números y mostrar cuál de los números es mayor.