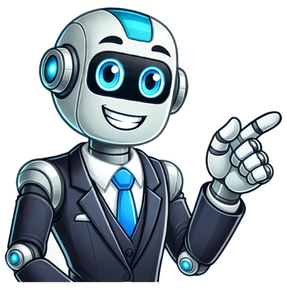
Click to prove you're human



Ejemplos de Colas Dinámicas en C: Definición según Autor, qué es, Concepto Ejemplos de conclusiones individuales: Definición según Autor, ¿qué es? Ejemplos de Globalización: Definición según Autor, qué es, Concepto Ejemplos de Automotivación Familiar: Definición según Autor, ¿qué es? Ejemplos de Diagrama: Definición según Autor, qué es, Concepto y Significado Ejemplos de organizaciones no gubernamentales internacionales: Definición Ejemplos de dones, talentos y habilidades: Definición según Autor, ¿qué es? Ejemplos de Ejercicios de Base de Daten: Definición según Autor, ¿qué es? Ejemplos de enlace ionico con nombre del compuesto: Definición según Autor Ejemplos de artistas con la voz grave: Definición según Autor, ¿qué es? Ejemplos de conectores de oposición: Definición según Autor, ¿qué es? Ejemplos de función fecha y hora en Excel 2010: Definición según Autor, qué Ejemplos de oraciones con verbos no copulativos: Definición según Autor Ejemplos de teoría de la partida doble: Definición según Autor, ¿qué es? Ejemplos de los tres estados fisicos del agua: Definición según Autor, qué Ejemplos de Diacronia y Sincronia en Historia: Definición según Autor, qué Ejemplos de Pleonasmi: Definición según Autor, qué es, Concepto Ejemplos de carátulas de grupo Isima: Definición según Autor, ¿qué es? Ejemplos de zonas de riesgo en la calle: Definición según Autor, ¿qué es? Ejemplos de mezclas efervescentes: Definición según Autor, qué es, Concepto Has escuchado alguna vez hablar de las estructuras secuenciales, si no lo sabes no te preocupes, en el día de hoy te traemos todo lo relacionado con este interesante tema, no te lo pierdas. Estructuras secuenciales En el mundo de la programación se consideran aquellas acciones donde se convierten en instrucciones, seguidas de otra secuencia. Las situaciones se presentan en forma de tareas que van consecutivas una después de otra: se puede decir entonces que dependen una de otra y se suceden de forma inmediata. En este sentido, la salida de una secuencia se convierte en la entrada de otra, originando la acción a través de sentencias, las cuales se suceden inmediatamente después y generan una operación o acción dentro de los recursos en los sistemas operativos. Las estructuras secuenciales se ejecutan en cada acción y lleva un orden respectivo, permitiendo a cada proceso generarse después de la culminación de otro, casi de forma inmediata. En el lenguaje programático sería de la siguiente manera, veamos el ejemplo: INPUT x INPUT y auxiliar= x x= y y= auxiliar PRINT x PRINT y Como vemos, es una secuencia de instrucciones que permita integrar los valores de «x» y «y», con ayuda de variables intermedias, la definición en términos entendibles sería la siguiente: Se guarda una copia del valor de x en auxiliar, se guarda el valor y en x, quien a su vez pierde su valor original, pero se mantiene una copia como contenido auxiliar, ese valor copia el valor auxiliar y lo convierte en el valor inicial de x. El resultado es el proceso de intercambio entre los valores de «x» y «y», con tres operaciones que deben llevar una secuencia definida para que la operación se produzca; de no colocar los comandos en el orden específico la secuencia se pierde y resulta inoperativa la acción. Componentes• Lo anterior nos lleva a considerar un algoritmo fácil de ejecutar, convirtiéndolo en un proceso cotidiano dentro de los procesos de ejecución de programas y comandos de un sistema. Para ello debe haber una serie de componentes que permitan su ejecución. Asignación El primer elemento está conformado por la asignación, la cual consiste en un paso de resultados a un área de la memoria, allí es reconocida con una variable y a su vez recibirá un valor. Esta asignación varía según ciertas especificaciones: -Simples o sencillas, es una acción de asignación donde se pasa un valor constante a una variable. -Contador, el valor se recibe igual pero pasa a ser constante en una variable. -Acumulador, se utiliza como un sumador para un proceso. -Trabajo, la asignación se recibe, y el resultado de la operación matemática, resulta de la inclusión de diversas variables. -Los formatos que deben utilizarse para realizar las asignaciones son las siguientes: < Variable >, Símbolos Son mandos que se envían a través de un dispositivo de salida, (Impresora, mouse, etc.) . A través de un mensaje, el cual resulta en una instrucción presentada en pantalla a través de un escrito entre comillas y con un contenido variable. Entrada de datos La entrada de datos se lleva a cabo a través de una lectura, la cual tiende a captar en el dispositivo de entrada como un teclado, un respectivo valor o dato; este se almacena en la variable que aparece inmediatamente después de la instrucción, y se presenta en el lenguaje de la siguiente manera: LEA < Variable >. Si deseas conocer más de estos temas te invitamos a leer el siguiente artículo Polimorfismo en programación, donde podrás aprender sobre otros contenidos similares. Proceso de variables Esta acción se encuentra dentro de las funciones de programación y se deriva de las estructuras secuenciales. Sirven para crear listas en el origen del algoritmo, sobre el total de los datos que posteriormente se utilizarán; de esta forma se lleva a cabo colocando el nombre de la variable, incluyendo su tipo. La declaración de variables incluye el contador, donde se puede colocar la edad si son esos datos los que se necesitan; se consideran entonces variables de tipo entero, pero si colocamos una declaración como salaraio_basico, se interpreta como un tipo de variable y será declarada como alfanumérica. Si al momento de hacer declaraciones constantes, existe la posibilidad de crear otros tipos, se debe indicar el respectivo valor. Los trabajos de programación con algoritmos no están determinados para realizar afirmaciones de datos. Asimismo, no se consideran constantes por motivos de facilidad, de manera que no es obligatorio declarar las variables en las estructuras secuenciales. Aplicación Estos procesos se llevan a cabo en algoritmos que sean más legibles y ordenados, así el programador se acostumbra a declararlas y mantener la secuencia, evitando interrupciones en las acciones. Por ejemplo, los lenguajes de programación como C + +, necesita de estas sentencias y declaraciones de variables, ya que de esa forma es como se ejecutan las funciones y los comandos mantienen la distribución y fluidez de las acciones. Como ejemplo, podemos decir que un algoritmo al cual se le asignan dos números y se pregunte con una variable relacionada con la suma, mostrará resultado de la operación entre ellos, es una acción sencilla pero implica la otorgación de declaraciones de variables. Otro ejemplo puede ser establecer el área de una figura geométrica otorgando las variables de altura y base. Comentario Final Los lenguajes de programación están diseñados para ser estructurados de manera eficiente según el tipo de software, brinda la oportunidad al ordenador realizar diversas tareas, sin embargo, las estructuras secuenciales se mantiene sin importar las versiones o actualizaciones, es una acción constante muy utilizada en las programaciones. Hemos terminado por hoy, deseamos que la información suministrada haya servido para ayudar a conocer más sobre las estructuras secuenciales, elemento importante dentro de los lenguajes de programación, los cuales deben ser estudiados con detenimiento. En programación, la estructura de secuencia es un tipo de estructura de control. También llamada estructura secuencial. La estructura de secuencia hace referencia al orden de ejecución de instrucciones que se hace de forma secuencial, o sea, una instrucción después de la otra. Las instrucciones se suceden de tal forma que el resultado de la anterior puede afectar la siguiente. Es el orden básico de ejecución en los lenguajes de programación, y su orden es de arriba a abajo. Es la base para la escritura de algoritmos. Veamos un ejemplo en pseudocódigo: a := 1 b := 7 r := a + b mostrar (r) r := a mostrar (r) Esto mostrará como resultado: 8 (primera invocación a la función mostrar) 1 (segunda invocación a la función mostrar) Como se puede apreciar, cada instrucción afecta las siguientes en un orden de ejecución secuencial de arriba a abajo, ejecutando una línea por vez. Estructura de secuencia: se realiza la instrucción 1, luego la instrucción 2 ... hasta la instrucción n Dentro de una estructura de secuencias podemos usar otras estructuras que la "rompen" con otro comportamiento. Las más típicas son: estructura de selección, y estructura de repetición. A su vez, dentro de cada una de estas estructuras de selección y de repetición hay estructuras de secuencia también. La estructura de secuencia es sumamente importante para la programación, ya que es a través de ella que se establece el orden de ejecución de las instrucciones. Además, es fundamental para la creación de algoritmos, que son utilizados en múltiples áreas. La estructura de selección es utilizada para tomar decisiones en base a ciertas condiciones establecidas, mientras que la estructura de repetición es utilizada para repetir un conjunto de instrucciones un número determinado de veces. En definitiva, la estructura de secuencia es la base sobre la cual se construyen las demás estructuras y es fundamental para el correcto funcionamiento de cualquier programa o algoritmo. Por tanto, es importante comprenderla y dominarla en profundidad para poder desarrollar soluciones eficientes y efectivas. La estructura de secuencia en programación se refiere al orden de ejecución de instrucciones de forma secuencial, una después de otra. Es el orden básico de ejecución en los lenguajes de programación y es la base para escribir algoritmos. Cada instrucción afecta la siguiente en un orden de ejecución de arriba a abajo. Puede ser interrumpida por otras estructuras como la de selección y repetición. La estructura de secuencia en programación es un tipo de estructura de control que determina el orden de ejecución de instrucciones de forma secuencial, una después de la otra. El resultado de una instrucción en la estructura de secuencia puede afectar al suceso de la siguiente, ya que el resultado de la instrucción anterior puede ser utilizado como entrada para la siguiente instrucción. El orden básico de ejecución en los lenguajes de programación es de arriba a abajo, es decir, se ejecuta una línea de código después de la otra, siguiendo el orden en el que fueron escritas. La estructura de secuencia es fundamental en la escritura de algoritmos, ya que permite establecer el orden en el que se deben ejecutar las instrucciones para lograr el resultado deseado. En el ejemplo de pseudocódigo dado, se realizan varias instrucciones en secuencia. Primero se realiza la suma de dos variables, luego se muestra el resultado. Después se asigna el valor de una variable a otra y se muestra nuevamente. Dentro de una estructura de secuencia en programación, se pueden utilizar otras estructuras que "rompen" este orden secuencial. Las más comunes son la estructura de selección (como el "if" condicional) y la estructura de repetición (como el "for" o "while"). Autor: Leandro Alegsa Actualizado: 18-06-2023 ¿Cómo citar este artículo? Alegsa, Leandro. (2023). Definición de Estructura de secuencia. Recuperado de Editor(a): jostin chañag Nivel: Ed. Básica y media Area Académica: Pensamiento Computacional Asignatura: Pensamiento Computacional Edad: Entre 11 a 12 años Número de Unidades: 3 Etiquetas: Pensamiento Computacional, Resolución de Problemas, Trabajo en Equipo Publicado el 18 Septiembre de 2024 En el mundo de la programación, las estructuras secuenciales juegan un papel fundamental en la creación de algoritmos eficientes y organizados. En este artículo, exploraremos en detalle qué son las estructuras secuenciales, cómo se utilizan en la programación y por qué son tan importantes para el desarrollo de software. ¡Acompáñanos en este viaje por el fascinante mundo de la programación secuencial!Índice En programación, las estructuras secuenciales son fundamentales para el desarrollo de algoritmos eficientes y funcionales.Las estructuras secuenciales permiten que las instrucciones se ejecuten en un orden específico, de manera secuencial, una tras otra. Esto es esencial para garantizar que el programa funcione de la manera deseada y produzca el resultado esperado.Algunas de las ventajas de utilizar estructuras secuenciales en programación son:Facilitan la comprensión del código.Permiten identificar y corregir errores de manera más sencilla.Ayudan a mantener un flujo lógico en la ejecución del programa.Además, las estructuras secuenciales son la base sobre la cual se construyen otras estructuras de control, como las estructuras condicionales y los bucles. Sin una secuencia lógica de instrucciones, el programa no podría funcionar correctamente y sería propenso a errores.Por lo tanto, es crucial que los programadores comprendan la importancia de utilizar estructuras secuenciales en sus programas y se aseguren de seguir un orden lógico en la ejecución de las instrucciones.Explorando el mundo de los secuenciales: su importancia y funcionamientoLeer Más 10 Ejemplos Creativos de Diagramas para InspirarteLos secuenciales son elementos fundamentales en el mundo de la programación, ya que permiten organizar y controlar el flujo de instrucciones de un programa de forma secuencial. Su importancia radica en que nos permiten ejecutar tareas de manera ordenada y predecible.¿Por qué son importantes los secuenciales?Los secuenciales son esenciales en la programación porque nos permiten definir el orden en el que se ejecutan las instrucciones de un programa. De esta manera, podemos controlar el flujo de datos y asegurarnos de que las acciones se realicen en el momento adecuado y en la secuencia correcta.¿Cómo funcionan los secuenciales?En el funcionamiento de los secuenciales, cada instrucción se ejecuta de manera secuencial, es decir, una detrás de la otra, siguiendo un orden establecido. Esto nos permite controlar el flujo de datos y la lógica de un programa de forma eficiente y organizada.Es importante tener en cuenta que los secuenciales son la base de la programación estructurada, ya que nos permiten dividir un problema en tareas más pequeñas y más manejables, facilitando así la escritura de programas más claros y fáciles de mantener.Recuerda practicar constantemente la creación de estructuras secuenciales en programación para mejorar tu habilidad y eficiencia en la resolución de problemas. Además, no dudes en consultar recursos adicionales y participar en comunidades de programadores para seguir aprendiendo y creciendo en este campo. ¡Éxito en tus proyectos futuros! ¡Hasta pronto!Si quieres ver otros artículos similares a Todo sobre las estructuras secuenciales en programación puedes visitar la categoría Diagramas de Flujo o revisar los siguientes artículos En programación, la estructura de secuencia es un tipo de estructura de control. También llamada estructura secuencial. La estructura de secuencia hace referencia al orden de ejecución de instrucciones que se hace de forma secuencial, o sea, una instrucción después de la otra. Las instrucciones se suceden de tal forma que el resultado de la anterior puede afectar la siguiente. Es el orden básico de ejecución en los lenguajes de programación, y su orden es de arriba a abajo. Es la base para la escritura de algoritmos. Veamos un ejemplo en pseudocódigo: a := 1 b := 7 r := a + b mostrar (r) r := a mostrar (r) Esto mostrará como resultado: 8 (primera invocación a la función mostrar) 1 (segunda invocación a la función mostrar) Como se puede apreciar, cada instrucción afecta las siguientes en un orden de ejecución secuencial de arriba a abajo, ejecutando una línea por vez. Estructura de secuencia: se realiza la instrucción 1, luego la instrucción 2 ... hasta la instrucción n Dentro de una estructura de secuencias podemos usar otras estructuras que la "rompen" con otro comportamiento. Las más típicas son: estructura de selección, y estructura de repetición. A su vez, dentro de cada una de estas estructuras de selección y de repetición hay estructuras de secuencia también. La estructura de secuencia es sumamente importante para la programación, ya que es a través de ella que se establece el orden de ejecución de las instrucciones. Además, es fundamental para la creación de algoritmos, que son utilizados en múltiples áreas. La estructura de selección es utilizada para tomar decisiones en base a ciertas condiciones establecidas, mientras que la estructura de repetición es utilizada para repetir un conjunto de instrucciones un número determinado de veces. En definitiva, la estructura de secuencia es la base sobre la cual se construyen las demás estructuras y es fundamental para el correcto funcionamiento de cualquier programa o algoritmo. Por tanto, es importante comprenderla y dominarla en profundidad para poder desarrollar soluciones eficientes y efectivas. La estructura de secuencia en programación se refiere al orden de ejecución de instrucciones de forma secuencial, una después de otra. Es el orden básico de ejecución en los lenguajes de programación y es la base para escribir algoritmos. Cada instrucción afecta la siguiente en un orden de ejecución de arriba a abajo. Puede ser interrumpida por otras estructuras como la de selección y repetición. La estructura de secuencia en programación es un tipo de estructura de control que determina el orden de ejecución de instrucciones de forma secuencial, una después de la otra. El resultado de una instrucción en la estructura de secuencia puede afectar al suceso de la siguiente, ya que el resultado de la instrucción anterior puede ser utilizado como entrada para la siguiente instrucción. El orden básico de ejecución en los lenguajes de programación es de arriba a abajo, es decir, se ejecuta una línea de código después de la otra, siguiendo el orden en el que fueron escritas. La estructura de secuencia es fundamental en la escritura de algoritmos, ya que permite establecer el orden en el que se deben ejecutar las instrucciones para lograr el resultado deseado. En el ejemplo de pseudocódigo dado, se realizan varias instrucciones en secuencia. Primero se realiza la suma de dos variables, luego se muestra el resultado. Después se asigna el valor de una variable a otra y se muestra nuevamente. Dentro de una estructura de secuencia en programación, se pueden utilizar otras estructuras que "rompen" este orden secuencial. Las más comunes son la estructura de selección (como el "if" condicional) y la estructura de repetición (como el "for" o "while"). Autor: Leandro Alegsa Actualizado: 18-06-2023 ¿Cómo citar este artículo? Alegsa, Leandro. (2023). Definición de Estructura de secuencia. Recuperado de Las estructuras secuenciales son una parte fundamental en la programación. Con ellas se pueden controlar la ejecución de un programa, definiendo el orden en que se deben realizar las instrucciones. Existen diferentes tipos de estructuras secuenciales y cada una de ellas tiene un uso específico. En este artículo, exploraremos los distintos tipos de estructuras secuenciales y su utilidad. Si estás interesado en aprender a programar, es importante que entiendas estos conceptos básicos. En TiposDe, encontrarás información detallada sobre diversos temas relacionados con la tecnología. ¡Sigue leyendo para descubrir los diferentes tipos de estructuras secuenciales! ¿De que hablaremos? La estructura secuencial es la más básica de todas. En ella, las instrucciones se ejecutan en el orden en que aparecen en el programa. Si no se utiliza otra estructura de control, el programa seguirá ejecutando las instrucciones secuencialmente. La estructura de selección se utiliza para ejecutar ciertas instrucciones únicamente si se cumple una condición determinada. Por ejemplo, si el usuario introduce un número menor que 10, se ejecutará una instrucción; si no, se ejecutará otra. La estructura de iteración se utiliza cuando se necesitan repetir una o varias instrucciones una cantidad determinada de veces. Por ejemplo, si queremos imprimir los números del 1 al 10, podemos utilizar una estructura de iteración para repetir la instrucción de impresión diez veces. La estructura de recursión se utiliza cuando una función se llama a sí misma una o varias veces, para resolver un problema. Por ejemplo, el cálculo del factorial de un número puede realizarse de forma recursiva. La estructura de subrutina se utiliza para dividir un programa en partes más pequeñas y manejables. Cada subrutina tiene un objetivo específico y puede ser llamada desde cualquier parte del programa donde sea necesario. Las funciones son una estructura similar a las subrutinas, pero con algunas diferencias importantes. En particular, las funciones pueden devolver un valor y tienen un tipo de retorno definido. Los bucles son estructuras que permiten repetir una o varias instrucciones mientras se cumple una condición. Existen diferentes tipos de bucles, como el bucle while, el bucle do-while y el bucle for. La estructura de manejo de excepciones se utiliza para manejar errores o situaciones imprevistas que pueden ocurrir durante la ejecución de un programa. Si una excepción es lanzada, se ejecutará el código correspondiente para manejar dicha excepción. La estructura de salto se utiliza para saltar a una determinada posición del programa. Por ejemplo, podemos utilizar un salto para salir de un bucle antes de que se cumpla la condición de salida. La estructura de asincronía se utiliza para permitir que una tarea se ejecute de forma independiente al resto del programa. Esto es útil en situaciones donde se necesitan realizar varias tareas al mismo tiempo. La estructura secuencial es la más básica de todas. La estructura de selección se utiliza para ejecutar ciertas instrucciones únicamente si se cumple una condición determinada. Las subrutinas son estructuras que permiten dividir un programa en partes más pequeñas y manejables. Los bucles permiten repetir una o varias instrucciones mientras se cumple una condición. La estructura de manejo de excepciones se utiliza para manejar errores o situaciones imprevistas que pueden ocurrir durante la ejecución de un programa. Las estructuras secuenciales son una parte fundamental en la programación. Cada una de ellas tiene una utilidad específica y permite controlar la ejecución de un programa de manera efectiva. Es importante que como programador, sepas cómo utilizar cada una de estas estructuras de forma apropiada, para lograr un programa eficiente y de calidad. En TiposDe, estamos comprometidos a brindarte información valiosa sobre tecnología y programación. Esperamos que este artículo te haya sido útil y te invitamos a seguir explorando nuestro sitio para aprender más sobre el apasionante mundo de la informática. Si tienes alguna opinión o duda sobre el tema de las estructuras secuenciales, no dudes en dejar un comentario. Nos encantaría escucharte y saber qué piensas sobre el uso de estas herramientas en la programación. Si quieres conocer otros artículos parecidos a Tipos de estructuras secuenciales puedes visitar la categoría General. ¿Ya conoces el concepto de estructura secuencial en programación? Si formas parte del mundo tecnológico, seguro que ya has escuchado o trabajado con diversas metodologías o estructuras de programación, como es el caso de la estructura secuencial. Esta se basa en un modelo simple y fácil de usar, donde una tarea sigue a la otra y esta, a su vez, a otra, de forma consecutiva. Para aprender más sobre algoritmos secuenciales, no dudes en seguir leyendo este artículo. La estructura secuencial o programación secuencial se entiende como una metodología que basa su funcionamiento en tener acciones o instrucciones que sigan a otras de forma secuencial. En este mecanismo se pueden presentar múltiples operaciones de inicio a fin, así como las operaciones de asignación o de cálculo, entre otras. La estructura secuencial o esquema secuencial en programación destaca también porque sus tareas se siguen de forma que la salida de una de estas actividades corresponde o hace las veces de entrada de la siguiente. Esto se da de forma sucesiva hasta que se finaliza el proceso en el sistema. La estructura secuencial es una de las estructuras de control básicas en programación y se caracteriza por la ejecución secuencial de instrucciones. Aquí están las características clave de la estructura secuencial: Secuencia de instrucciones: Las instrucciones se ejecutan en un orden específico, de arriba a abajo, una tras otra, sin bifurcaciones ni saltos. La ejecución sigue un flujo lineal. Un solo punto de entrada y salida: La ejecución comienza en un solo punto de entrada y continúa de manera secuencial hasta alcanzar un punto de salida. No hay bifurcaciones ni bucles en la estructura secuencial básica. Ejecución unidireccional: La ejecución del programa sigue una dirección unidireccional, avanzando de una instrucción a la siguiente sin retroceder ni desviarse. Fácil lectura y mantenimiento: La estructura secuencial es fácil de leer y comprender, ya que refleja el flujo natural del pensamiento humano. Esto facilita el mantenimiento y la depuración del código. Instrucciones ejecutadas una única vez: Cada instrucción dentro de la estructura secuencial se ejecuta exactamente una vez. No hay ciclos o repeticiones automáticas en esta estructura básica. Ejecución lineal: El control del programa se mueve de una instrucción a la siguiente en un flujo lineal. No hay bifurcaciones condicionales ni ciclos que alteren el orden secuencial. Aplicación de operaciones básicas: Las instrucciones dentro de la estructura secuencial suelen consistir en operaciones básicas, como asignaciones, cálculos y llamadas a funciones, sin introducir estructuras de control más complejas. Simplicidad estructural: La estructura secuencial es la más simple de todas las estructuras de control. Aunque es limitada en términos de control de flujo, es esencial para construir bloques fundamentales de lógica en programas. La estructura secuencial es fundamental en la programación, especialmente en el desarrollo de algoritmos y scripts simples. Sin embargo, en la mayoría de los programas más complejos, se combinan diversas estructuras de control, como estructuras de decisión y bucles, para crear una lógica más avanzada y flexible. En este artículo has podido conocer todo lo necesario acerca de la estructura secuencial en programación, así como cuáles son sus principales características. No dudes en dar un paso más para continuar formándote y destacar en el sector IT y matricúlate en nuestro Desarrollo de Apps Móviles Full Stack Bootcamp. Con la guía de los expertos que te formarán en este curso intensivo, podrás adquirir los conocimientos teóricos y prácticos necesarios para convertirte en un verdadero experto del mundo tecnológico. ¡Anímate a formar parte de este bootcamp para darle un impulso a tu vida! Has escuchado alguna vez hablar de las estructuras secuenciales, si no lo sabes no te preocupes, en el día de hoy te traemos todo lo relacionado con este interesante tema, no te lo pierdas. Estructuras secuenciales En el mundo de la programación se consideran aquellas acciones donde se convierten en instrucciones, seguidas de otra secuencia. Las situaciones se presentan en forma de tareas que van consecutivas una después de otra: se puede decir entonces que dependen una de otra y se suceden de forma inmediata. En este sentido, la salida de una secuencia se convierte en la entrada de otra, originando la acción a través de sentencias, las cuales se suceden inmediatamente después y generan una operación o acción dentro de los recursos en los sistemas operativos. Las estructuras secuenciales se ejecutan en cada acción y lleva un orden respectivo, permitiendo a cada proceso generarse después de la culminación de otro, casi de forma inmediata. En el lenguaje programático sería de la siguiente manera, veamos el ejemplo: INPUT x INPUT y auxiliar= x x= y y= auxiliar PRINT x PRINT y Como vemos, es una secuencia de instrucciones que permita integrar los valores de «x» y «y», con ayuda de variables intermedias, la definición en términos entendibles sería la siguiente: Se guarda una copia del valor de x en auxiliar, se guarda el valor y en x, quien a su vez pierde su valor original, pero se mantiene una copia como contenido auxiliar, ese valor copia el valor auxiliar y lo convierte en el valor inicial de x. El resultado es el proceso de intercambio entre los valores de «x» y «y», con tres operaciones que deben llevar una secuencia definida para que la operación se produzca; de no colocar los comandos en el orden específico la secuencia se pierde y resulta inoperativa la acción. Componentes• Lo anterior nos lleva a considerar un algoritmo fácil de ejecutar, convirtiéndolo en un proceso cotidiano dentro de los procesos de ejecución de programas y comandos de un sistema. Para ello debe haber una serie de componentes que permitan su ejecución. Asignación El primer elemento está conformado por la asignación, la cual consiste en un paso de resultados a un área de la memoria, allí es reconocida con una variable y a su vez recibirá un valor. Esta asignación varía según ciertas especificaciones: -Simples o sencillas, es una acción de asignación donde se pasa un valor constante a una variable. -Contador, el valor se recibe igual pero pasa a ser constante en una variable. -Acumulador, se utiliza como un sumador para un proceso. -Trabajo, la asignación se recibe, y el resultado de la operación matemática, resulta de la inclusión de diversas variables. -Los formatos que deben utilizarse para realizar las asignaciones son las siguientes: < Variable >, Símbolos Son mandos que se envían a través de un dispositivo de salida, (Impresora, mouse, etc.) . A través de un mensaje, el cual resulta en una instrucción presentada en pantalla a través de un escrito entre comillas y con un contenido variable. Entrada de datos La entrada de datos se lleva a cabo a través de una lectura, la cual tiende a captar en el dispositivo de entrada como un teclado, un respectivo valor o dato; este se almacena en la variable que aparece inmediatamente después de la instrucción, y se presenta en el lenguaje de la siguiente manera: LEA < Variable >. Si deseas conocer más de estos temas te invitamos a leer el siguiente artículo Polimorfismo en programación, donde podrás aprender sobre otros contenidos similares. Proceso de variables Esta acción se encuentra dentro de las funciones de programación y se deriva de las estructuras secuenciales. Sirven para crear listas en el origen del algoritmo, sobre el total de los datos que posteriormente se utilizarán; de esta forma se lleva a cabo colocando el nombre de la variable, incluyendo su tipo. La declaración de variables incluye el contador, donde se puede colocar la edad si son esos datos los que se necesitan; se consideran entonces variables de tipo entero, pero si colocamos una declaración como salaraio_basico, se interpreta como un tipo de variable y será declarada como alfanumérica. Si al momento de hacer declaraciones constantes, existe la posibilidad de crear otros tipos, se debe indicar el respectivo valor. Los trabajos de programación con algoritmos no están determinados para realizar afirmaciones de datos. Asimismo, no se consideran constantes por motivos de facilidad, de manera que no es obligatorio declarar las variables en las estructuras secuenciales. Aplicación Estos procesos se llevan a cabo en algoritmos que sean más legibles y ordenados, así el programador se acostumbra a declararlas y mantener la secuencia, evitando interrupciones en las acciones. Por ejemplo, los lenguajes de programación como C + +, necesita de estas sentencias y declaraciones de variables, ya que de esa forma es como se ejecutan las funciones y los comandos mantienen la distribución y fluidez de las acciones. Como ejemplo, podemos decir que un algoritmo al cual se le asignan dos números y se pregunte con una variable relacionada con la suma, mostrará resultado de la operación entre ellos, es una acción sencilla pero implica la otorgación de declaraciones de variables. Otro ejemplo puede ser establecer el área de una figura geométrica otorgando las variables de altura y base. Comentario Final Los lenguajes de programación están diseñados para ser estructurados de manera eficiente según el tipo de software, brinda la oportunidad al ordenador realizar diversas tareas, sin embargo, las estructuras secuenciales se mantiene sin importar las versiones o actualizaciones, es una acción constante muy utilizada en las programaciones. Hemos terminado por hoy, deseamos que la información suministrada haya servido para ayudar a conocer más sobre las estructuras secuenciales, elemento importante dentro de los lenguajes de programación, los cuales deben ser estudiados con detenimiento. Has escuchado alguna vez hablar de las estructuras secuenciales, si no lo sabes no te preocupes, en el día de hoy te traemos todo lo relacionado con este interesante tema, no te lo pierdas. Estructuras secuenciales En el mundo de la programación se consideran aquellas acciones donde se convierten en instrucciones, seguidas de otra secuencia. Las situaciones se presentan en forma de tareas que van consecutivas una después de otra: se puede decir entonces que dependen una de otra y se suceden de forma inmediata. En este sentido, la salida de una secuencia se convierte en la entrada de otra, originando la acción a través de sentencias, las cuales se suceden inmediatamente después y generan una operación o acción dentro de los recursos en los sistemas operativos. Las estructuras secuenciales se ejecutan en cada acción y lleva un orden respectivo, permitiendo a cada proceso generarse después de la culminación de otro, casi de forma inmediata. En el lenguaje programático sería de la siguiente manera, veamos el ejemplo: INPUT x INPUT y auxiliar= x x= y y= auxiliar PRINT x PRINT y Como vemos, es una secuencia de instrucciones que permita integrar los valores de «x» y «y», con ayuda de variables intermedias, la definición en términos entendibles sería la siguiente: Se guarda una copia del valor de x en auxiliar, se guarda el valor y en x, quien a su vez pierde su valor original, pero se mantiene una copia como contenido auxiliar, ese valor copia el valor auxiliar y lo convierte en el valor inicial de x. El resultado es el proceso de intercambio entre los valores de «x» y «y», con tres operaciones que deben llevar una secuencia definida para que la operación se produzca; de no colocar los comandos en el orden específico la secuencia se pierde y resulta inoperativa la acción. Componentes• Lo anterior nos lleva a considerar un algoritmo fácil de ejecutar, convirtiéndolo en un proceso cotidiano dentro de los procesos de ejecución de programas y comandos de un sistema. Para ello debe haber una serie de componentes que permitan su ejecución. Asignación El primer elemento está conformado por la asignación, la cual consiste en un paso de resultados a un área de la memoria, allí es reconocida con una variable y a su vez recibirá un valor. Esta asignación varía según ciertas especificaciones: -Simples o sencillas, es una acción de asignación donde se pasa un valor constante a una variable. -Contador, el valor se recibe igual pero pasa a ser constante en una variable. -Acumulador, se utiliza como un sumador para un proceso. -Trabajo, la asignación se recibe, y el resultado de la operación matemática, resulta de la inclusión de diversas variables. -Los formatos que deben utilizarse para realizar las asignaciones son las siguientes: < Variable >, Símbolos Son mandos que se envían a través de un dispositivo de salida, (Impresora, mouse, etc.) . A través de un mensaje, el cual resulta en una instrucción presentada en pantalla a través de un escrito entre comillas y con un contenido variable. Entrada de datos La entrada de datos se lleva a cabo a través de una lectura, la cual tiende a captar en el dispositivo de entrada como un teclado, un respectivo valor o dato; este se almacena en la variable que aparece inmediatamente después de la instrucción, y se presenta en el lenguaje de la siguiente manera: LEA < Variable >. Si deseas conocer más de estos temas te invitamos a leer el siguiente artículo Polimorfismo en programación, donde podrás aprender sobre otros contenidos similares. Proceso de variables Esta acción se encuentra dentro de las funciones de programación y se deriva de las estructuras secuenciales. Sirven para crear listas en el origen del algoritmo, sobre el total de los datos que posteriormente se utilizarán; de esta forma se lleva a cabo colocando el nombre de la variable, incluyendo su tipo. La declaración de variables incluye el contador, donde se puede colocar la edad si son esos datos los que se necesitan; se consideran entonces variables de tipo entero, pero si colocamos una declaración como salaraio_basico, se interpreta como un tipo de variable y será declarada como alfanumérica. Si al momento de hacer declaraciones constantes, existe la posibilidad de crear otros tipos, se debe indicar el respectivo valor. Los trabajos de programación con algoritmos no están determinados para realizar afirmaciones de datos. Asimismo, no se consideran constantes por motivos de facilidad, de manera que no es obligatorio declarar las variables en las estructuras secuenciales. Aplicación Estos procesos se llevan a cabo en algoritmos que sean más legibles y ordenados, así el programador se acostumbra a declararlas y mantener la secuencia, evitando interrupciones en las acciones. Por ejemplo, los lenguajes de programación como C + +, necesita de estas sentencias y declaraciones de variables, ya que de esa forma es como se ejecutan las funciones y los comandos mantienen la distribución y fluidez de las acciones. Como ejemplo, podemos decir que un algoritmo al cual se le asignan dos números y se pregunte con una variable relacionada con la suma, mostrará resultado de la operación entre ellos, es una acción sencilla pero implica la otorgación de declaraciones de variables. Otro ejemplo puede ser establecer el área de una figura geométrica otorgando las variables de altura y base. Comentario Final Los lenguajes de programación están diseñados para ser estructurados de manera eficiente según el tipo de software, brinda la oportunidad al ordenador realizar diversas tareas, sin embargo, las estructuras secuenciales se mantiene sin importar las versiones o actualizaciones, es una acción constante muy utilizada en las programaciones. Hemos terminado por hoy, deseamos que la información suministrada haya servido para ayudar a conocer más sobre las estructuras secuenciales, elemento importante dentro de los lenguajes de programación, los cuales deben ser estudiados con detenimiento. En este capítulo aprenderás a distinguir y usar 3 estructuras secuenciales: la asignación, la lectura o entrada de datos, la salida de datos. Definicion La estructura secuencial es aquella en la que una acción (instrucción) sigue a otra en secuencia. Las tareas se suceden de tal modo que la salida de una es la entrada de la siguiente y así sucesivamente hasta el fin del proceso. Una estructura secuencial se representa de la siguiente forma: accion 1 accion 2 . . accion n Asignación La asignación consiste, en el paso de valores o resultados a una variable. Las asignaciones se pueden clasificar de la siguiente forma: Simples Consiste en pasar un valor constante a una variable (a = 15) De trabajo Donde se recibe el resultado de una operación que involucra varias variables o valores (a = c + b * 2 / 4). Contador Consiste en usar una variable para contar el número de veces que se realiza un proceso (a = a + 1) Acumulador Consiste en usar una variable para acumular un valor en un proceso que se puede repetir varias veces (a = a + b). Lectura La lectura consiste en recibir desde un dispositivo de entrada (generalmente, el teclado y el mouse) un valor. Esta operación se representa en un pseudo código como sigue: leer valor Donde valor es el nombre de la variable que recibirá el valor. Escritura Consiste en mandar por un dispositivo de salida (generalmente, el monitor) un resultado o mensaje. Este proceso se representa en un pseudocódigo como sigue: escribir "El resultado es:", resultado Donde "El resultado es:" es un mensaje que se desea aparezca en la pantalla y resultado es una variable que contiene un valor. Problemas Secuenciales Suponga que un individuo desea invertir su capital en un banco y desea saber cuanto dinero ganara después de un mes, si el banco paga a razón de 2% mensual. inicio escribir "Proporcione el capital invertido: " leer capitalInvertido ganancia = capitalInvertido * 0.02 escribir "Su ganancia sera de: ", ganancia fin Una tienda ofrece un descuento del 15% sobre el total de la compra y un cliente desea saber cuanto deberá pagar finalmente por su compra. inicio escribir "Cuanto compraste?" leer totalCompra descuento = totalCompra * 0.15 pagoFinal = totalCompra - descuento escribir "Vas a pagar ", pagoFinal fin Realizar un algoritmo que calcule la edad de una persona. inicio escribir "En que anio naciste?" leer anioNacimiento escribir "En que anio estamos?" leer anioActual edad = anioActual - anioNacimiento escribir "Tienes", edad, "anios" fin Un vendedor recibe un sueldo base mas un 10% extra por comisión de sus ventas, el vendedor desea saber cuanto dinero obtendrá por concepto de comisiones por las tres ventas que realiza en el mes y el total que recibirá en el mes tomando en cuenta su sueldo base y comisiones. inicio escribir "De cuanto es tu sueldo base? " leer sueldoBase escribir "De cuanto fue tu primer venta? " leer ventaUno escribir "De cuanto fue tu segunda venta? " leer ventaDos escribir "De cuanto fue tu tercer venta? " leer ventaTres totalVentas = ventaUno + ventaDos + ventaTres comisiones = totalVentas * 0.10 sueldoFinal = sueldoBase + comisiones escribir "Vas a recibir de comisiones: ",comisiones escribir "Tu sueldo final es de: ", sueldoFinal fin Un maestro desea saber que porcentaje de hombres y que porcentaje de mujeres hay en un grupo de estudiantes. inicio escribir "Proporcione la cantidad de mujeres:" leer numeroMujeres escribir "Proporcione la cantidad de hombres:" leer numeroHombres totalAlumnos = numeroMujeres + numeroHombres porcentajeMujeres = numeroMujeres * 100 / totalAlumnos porcentajeHombres = numeroHombres * 100 / totalAlumnos escribir "El porcentaje de mujeres es de: ", porcentajeMujeres escribir "El porcentaje de hombres es de: ", porcentajeHombres fin Un alumno desea saber cual será su calificación final en la materia de Algoritmos. Dicha calificación se compone de los siguientes porcentajes: 55% del promedio de sus tres calificaciones parciales. 30% de la calificación del examen final. 15% de la calificación de un trabajo final. inicio escribir "Escriba la calificacion del primer examen: " leer calificacion1 escribir "Escriba la calificacion del segundo examen: " leer calificacion2 escribir "Escriba la calificacion del tercer examen: " leer calificacion3 escribir "Escriba la calificacion del examen final: " leer calificacionExamenFinal escribir "Escriba la calificacion del trabajo final: " leer calificacionTrabajoFinal promedioParciales = (calificacion1 + calificacion2 + calificacion3)/3 porcentajePromedio = promedioParciales * 0.55 porcentajeExamenFinal = calificacionExamenFinal * 0.30 porcentajeTrabajoFinal = calificacionTrabajoFinal * 0.15 calificacionFinal = porcentajePromedio + porcentajeExamenFinal + porcentajeTrabajoFinal escribir "Tu calificacion final es: ", calificacionFinal fin Problemas Propuestos Dada un cantidad en pesos, obtener la equivalencia en dólares, asumiendo que la unidad cambiaria es un dato desconocido. La presión, el volumen y la temperatura de una masa de aire se relacionan por la fórmula: masa = (presion * volumen) / (0.37 * (temperatura + 460)) Calcular el numero de pulsaciones que una persona debe tener por cada 10 segundos de ejercicio, si la formula es: numeroPulsaciones = (220 - edad) / 10 Calcular el nuevo salario de un obrero si obtuvo un incremento del 25% sobre su salario anterior. En un hospital existen tres áreas: Ginecología, Pediatría, Traumatología. El presupuesto anual del hospital se reparte conforme a la siguiente tabla: Área Porcentaje del presupuesto Ginecología 40% Traumatología 30% Pediatría 30% Obtener la cantidad de dinero que recibirá cada área, para cualquier monto presupuestal. El dueño de una tienda compra un artículo a un precio determinado. Obtener el precio en que lo debe vender para obtener una ganancia del 30%. Todos los lunes, miércoles y viernes, una persona corre la misma ruta y cronometra los tiempos obtenidos. Determinar el tiempo promedio que la persona tarda en recorrer la ruta en una semana cualquiera. Tres personas deciden invertir su dinero para fundar una empresa. Cada una de ellas invierte una cantidad distinta. Obtener el porcentaje que cada quien invierte con respecto a la cantidad total invertida. Un alumno desea saber cual será su promedio general en las tres materias más difíciles que cursa y cual será el promedio que obtendrá en cada una de ellas. Estas materias se evalúan como se muestra a continuación: La calificación de Matemáticas se obtiene de la siguiente manera: Examen 90% Promedio de tareas 10% En esta materia se pidió un total de tres tareas. La calificación de Física se obtiene de la sig. manera: Examen 80% Promedio de tareas 20% En esta materia se pidió un total de dos tareas. La calificación de Química se obtiene de la sig. manera: Examen 85% Promedio de tareas 15% En esta materia se pidió un promedio de tres tareas. Has escuchado alguna vez hablar de las estructuras secuenciales, si no lo sabes no te preocupes, en el día de hoy te traemos todo lo relacionado con este interesante tema, no te lo pierdas. Estructuras secuenciales En el mundo de la programación se consideran aquellas acciones donde se convierten en instrucciones, seguidas de otra secuencia. Las situaciones se presentan en forma de tareas que van consecutivas una después de otra: se puede decir entonces que dependen una de otra y se suceden de forma inmediata. En este sentido, la salida de una secuencia se convierte en la entrada de otra, originando la acción a través de sentencias, las cuales se suceden inmediatamente después y generan una operación o acción dentro de los recursos en los sistemas operativos. Las estructuras secuenciales se ejecutan en cada acción y lleva un orden respectivo, permitiendo a cada proceso generarse después de la culminación de otro, casi de forma inmediata. En el lenguaje programático sería de la siguiente manera, veamos el ejemplo: INPUT x INPUT y auxiliar= x x= y y= auxiliar PRINT x PRINT y Como vemos, es una secuencia de instrucciones que permita integrar los valores de «x» y «y», con ayuda de variables intermedias, la definición en términos entendibles sería la siguiente: Se guarda una copia del valor de x en auxiliar, se guarda el valor y en x, quien a su vez pierde su valor original, pero se mantiene una copia como contenido auxiliar, ese valor copia el valor auxiliar y lo convierte en el valor inicial de x. El resultado es el proceso de intercambio entre los valores de «x» y «y», con tres operaciones que deben llevar una secuencia definida para que la operación se produzca; de no colocar los comandos en el orden específico la secuencia se pierde y resulta inoperativa la acción. Componentes• Lo anterior nos lleva a considerar un algoritmo fácil de ejecutar, convirtiéndolo en un proceso cotidiano dentro de los procesos de ejecución de programas y comandos de un sistema. Para ello debe haber una serie de componentes que permitan su ejecución. Asignación El primer elemento está conformado por la asignación, la cual consiste en un paso de resultados a un área de la memoria, allí es reconocida con una variable y a su vez recibirá un valor. Esta asignación varía según ciertas especificaciones: -Simples o sencillas, es una acción de asignación donde se pasa un valor constante a una variable. -Contador, el valor se recibe igual pero pasa a ser constante en una variable. -Acumulador, se utiliza como un sumador para un proceso. -Trabajo, la asignación se recibe, y el resultado de la operación matemática, resulta de la inclusión de diversas variables. -Los formatos que deben utilizarse para realizar las asignaciones son las siguientes: < Variable >, Símbolos Son mandos que se envían a través de un dispositivo de salida, (Impresora, mouse, etc.) . A través de un mensaje, el cual resulta en una instrucción presentada en pantalla a través de un escrito entre comillas y con un contenido variable. Entrada de datos La entrada de datos se lleva a cabo a través de una lectura, la cual tiende a captar en el dispositivo de entrada como un teclado, un respectivo valor o dato; este se almacena en la variable que aparece inmediatamente después de la instrucción, y se presenta en el lenguaje de la siguiente manera: LEA < Variable >. Si deseas conocer más de estos temas te invitamos a leer el siguiente artículo Polimorfismo en programación, donde podrás aprender sobre otros contenidos similares. Proceso de variables Esta acción se encuentra dentro de las funciones de programación y se deriva de las estructuras secuenciales. Sirven para crear listas en el origen del algoritmo, sobre el total de los datos que posteriormente se utilizarán; de esta forma se lleva a cabo colocando el nombre de la variable, incluyendo su tipo. La declaración de variables incluye el contador, donde se puede colocar la edad si son esos datos los que se necesitan; se consideran entonces variables de tipo entero, pero si colocamos una declaración como salaraio_basico, se interpreta como un tipo de variable y será declarada como alfanumérica. Si al momento de hacer declaraciones constantes, existe la posibilidad de crear otros tipos, se debe indicar el respectivo valor. Los trabajos de programación con algoritmos no están determinados para realizar afirmaciones de datos. Asimismo, no se consideran constantes por motivos de facilidad, de manera que no es obligatorio declarar las variables en las estructuras secuenciales. Aplicación Estos procesos se llevan a cabo en algoritmos que sean más legibles y ordenados, así el programador se acostumbra a declararlas y mantener la secuencia, evitando interrupciones en las acciones. Por ejemplo, los lenguajes de programación como C + +, necesita de estas sentencias y declaraciones de variables, ya que de esa forma es como se ejecutan las funciones y los comandos mantienen la distribución y fluidez de las acciones. Como ejemplo, podemos decir que un algoritmo al cual se le asignan dos números y se pregunte con una variable relacionada con la suma, mostrará resultado de la operación entre ellos, es una acción sencilla pero implica la otorgación de declaraciones de variables. Otro ejemplo puede ser establecer el área de una figura geométrica otorgando las variables de altura y base. Comentario Final Los lenguajes de programación están diseñados para ser estructurados de manera eficiente según el tipo de software, brinda la oportunidad al ordenador realizar diversas tareas, sin embargo, las estructuras secuenciales se mantiene sin importar las versiones o actualizaciones, es una acción constante muy utilizada en las programaciones. Hemos terminado por hoy, deseamos que la información suministrada haya servido para ayudar a conocer más sobre las estructuras secuenciales, elemento importante dentro de los lenguajes de programación, los cuales deben ser estudiados con detenimiento. Has escuchado alguna vez hablar de las estructuras secuenciales, si no lo sabes no te preocupes, en el día de hoy te traemos todo lo relacionado con este interesante tema, no te lo pierdas. Estructuras secuenciales En el mundo de la programación se consideran aquellas acciones donde se convierten en instrucciones, seguidas de otra secuencia. Las situaciones se presentan en forma de tareas que van consecutivas una después de otra: se puede decir entonces que dependen una de otra y se suceden de forma inmediata. En este sentido, la salida de una secuencia se convierte en la entrada de otra, originando la acción a través de sentencias, las cuales se suceden inmediatamente después y generan una operación o acción dentro de los recursos en los sistemas operativos. Las estructuras secuenciales se ejecutan en cada acción y lleva un orden respectivo, permitiendo a cada proceso generarse después de la culminación de otro, casi de forma inmediata. En el lenguaje programático sería de la siguiente manera, veamos el ejemplo: INPUT x INPUT y auxiliar= x x= y y= auxiliar PRINT x PRINT y Como vemos, es una secuencia de instrucciones que permita integrar los valores de «x» y «y», con ayuda de variables intermedias, la definición en términos entendibles sería la siguiente: Se guarda una copia del valor de x en auxiliar, se guarda el valor y en x, quien a su vez pierde su valor original, pero se mantiene una copia como contenido auxiliar, ese valor copia el valor auxiliar y lo convierte en el valor inicial de x. El resultado es el proceso de intercambio entre los valores de «x» y «y», con tres operaciones que deben llevar una secuencia definida para que la operación se produzca; de no colocar los comandos en el orden específico la secuencia se pierde y resulta inoperativa la acción. Componentes• Lo anterior nos lleva a considerar un algoritmo fácil de ejecutar, convirtiéndolo en un proceso cotidiano dentro de los procesos de ejecución de programas y comandos de un sistema. Para ello debe haber una serie de componentes que permitan su ejecución. Asignación El primer elemento está conformado por la asignación, la cual consiste en un paso de resultados a un área de la memoria, allí es reconocida con una variable y a su vez recibirá un valor. Esta asignación varía según ciertas especificaciones: -Simples o sencillas, es una acción de asignación donde se pasa un valor constante a una variable. -Contador, el valor se recibe igual pero pasa a ser constante en una variable. -Acumulador, se utiliza como un sumador para un proceso. -Trabajo, la asignación se recibe, y el resultado de la operación matemática, resulta de la inclusión de diversas variables. -Los formatos que deben utilizarse para realizar las asignaciones son las siguientes: < Variable >, Símbolos Son mandos que se envían a través de un dispositivo de salida, (Impresora, mouse, etc.) . A través de un mensaje, el cual resulta en una instrucción presentada en pantalla a través de un escrito entre comillas y con un contenido variable. Entrada de datos La entrada de datos se lleva a cabo a través de una lectura, la cual tiende a captar en el dispositivo de entrada como un teclado, un respectivo valor o dato; este se almacena en la variable que aparece inmediatamente después de la instrucción, y se presenta en el lenguaje de la siguiente manera: LEA < Variable >. Si deseas conocer más de estos temas te invitamos a leer el siguiente artículo Polimorfismo en programación, donde podrás aprender sobre otros contenidos similares. Proceso de variables Esta acción se encuentra dentro de las funciones de programación y se deriva de las estructuras secuenciales. Sirven para crear listas en el origen del algoritmo, sobre el total de los datos que posteriormente se utilizarán; de esta forma se lleva a cabo colocando el nombre de la variable, incluyendo su tipo. La declaración de variables incluye el contador, donde se puede colocar la edad si son esos datos los que se necesitan; se consideran entonces variables de tipo entero, pero si colocamos una declaración como salaraio_basico, se interpreta como un tipo de variable y será declarada como alfanumérica. Si al momento de hacer declaraciones constantes, existe la posibilidad de crear otros tipos, se debe indicar el respectivo valor. Los trabajos de programación con algoritmos no están determinados para realizar afirmaciones de datos. Asimismo, no se consideran constantes por motivos de facilidad, de manera que no es obligatorio declarar las variables en las estructuras secuenciales. Aplicación Estos procesos se llevan a cabo en algoritmos que sean más legibles y ordenados, así el programador se acostumbra a declararlas y mantener la secuencia, evitando interrupciones en las acciones. Por ejemplo, los lenguajes de programación como C + +, necesita de estas sentencias y declaraciones de variables, ya que de esa forma es como se ejecutan las funciones y los comandos mantienen la distribución y fluidez de las acciones. Como ejemplo, podemos decir que un algoritmo al cual se le asignan dos números y se pregunte con una variable relacionada con la suma, mostrará resultado de la operación entre ellos, es una acción sencilla pero implica la otorgación de declaraciones de variables. Otro ejemplo puede ser establecer el área de una figura geométrica otorgando las variables de altura y base. Comentario Final Los lenguajes de programación están diseñados para ser estructurados de manera eficiente según el tipo de software, brinda la oportunidad al ordenador realizar diversas tareas, sin embargo, las estructuras secuenciales se mantiene sin importar las versiones o actualizaciones, es una acción constante muy utilizada en las programaciones. Hemos terminado por hoy, deseamos que la información suministrada haya servido para ayudar a conocer más sobre las estructuras secuenciales, elemento importante dentro de los lenguajes de programación, los cuales deben ser estudiados con detenimiento.