

Actividad | #1 | Algoritmos

Introducción al Desarrollo de Software

Ingeniería en Desarrollo de Software



TUTOR: Sandra Luz Lara Dévora

ALUMNO: Alan Giovanni Nieto Becerril

FECHA: 12/04/25

Índice

Introducción 3

Descripción 4

Justificación 5

Desarrollo 6

Conclusión 9

Referencias 10

Introducción

En la vida cotidiana utilizamos algoritmos para realizar diversas tareas, incluso de forma inconsciente. Los encontramos al seguir recetas, al realizar operaciones matemáticas cuando realizamos alguna compra, al manejar para ir al trabajo, al seleccionar la ropa que usaremos cada día y vestírnos, entre otros. Los algoritmos son conjuntos de operaciones cuyo objetivo es resolver un problema siguiendo instrucciones lógicas. En este procedimiento se utiliza una serie de pasos precisos y detallados. Algunas otras características de los algoritmos es que deben ser finitos y ordenados. Son un componente esencial para desarrollar aplicaciones. Los algoritmos tienen la siguiente estructura: 1) Entrada. Esto significa que se requiere una fuente de datos para que los procesos puedan comenzar; 2) Proceso. Se define como la ejecución paso a paso de instrucciones; 3) Salida. Es el resultado final, cuando los datos de entrada pasan a través del proceso, se transforman y deben dar respuesta al problema inicial.

Descripción

En esta actividad se trabajará con tres tipos de calculadoras diferentes que deben implementarse en colegios. Únicamente desarrollaremos el algoritmo de cada calculadora y en las siguientes actividades se realizarán los diagramas de flujo y la programación en lenguaje C.

La primera calculadora se llamará ‘primos’ y debe identificar cuando un número ingresado por el usuario sea un número primo. Debemos considerar que un número primo es un número entero que solo es divisible por sí mismo y por el 1.

La segunda calculadora se llamará ‘par/impar’ y a esta se le darán 10 números, de los cuáles indicará cuales son pares y cuales son impares. Un número par es divisible entre 2, mientras que uno impar no lo es.

La tercera calculadora se llamará ‘al revés’. Esta calculadora debe recibir un número entero de 4 dígitos y a través de operaciones matemáticas debe invertir el orden de los dígitos.

Justificación

El uso de los algoritmos en el desarrollo de software es esencial para comprender de manera detallada y concisa lo que se quiere codificar. Es una herramienta que nos sirve para tener certeza de que la problemática que queremos resolver si tendrá una respuesta. En los casos presentados en esta actividad, utilizar algoritmos sirvió para desglosar todas las instrucciones que debemos seguir por pasos para que las calculadoras funcionen correctamente. Estas instrucciones serán representadas en los diagramas de flujo y al momento de programarlos, utilizando ya una sintaxis de un lenguaje de programación. En general, es una buena práctica para ayudar a estructurar el pensamiento lógico que queremos plasmar. Además de que nos ayudará a identificar errores lógicos en específico. Tiene la ventaja de que lo usamos con lenguaje natural, no lenguaje de programación, lo que facilita su entendimiento a cualquier persona. Existen otras herramientas como los pseudocódigos, de los cuales nos podemos apoyar para expresar la lógica que el algoritmo debe seguir.

Desarrollo

Calculadora 'Primos':

Inicio

***Solicitar un número para determinar si es primo o no**

Escribir "Ingrese un número entero"

Leer el número ingresado

***Inicializar variables**

iteracion = 1

divisionResiduoCero = 0

***Contar cuantos divisores tiene el número con un bucle**

Mientras iteracion <= numeroingresado Hacer

Si(numeroingresado % iteración == 0) Entonces

divisionResiduoCero = divisionResiduoCero + 1

FinSi

iteracion = iteracion + 1

Fin Mientras

***Determinar si es primo, si divisionResiduoCero es igual a 2**

Si (divisionResiduoCero == 2) Entonces

Escribir "El número que ingresaste es un número primo"

SiNo

Escribir "El número que ingresaste no es un número primo"

FinSi

Fin

Calculadora 'Par/Impar':

Inicio

*Inicializar la variable num que servirá como iterador

num = 0

Repetir

num = num + 1

*Solicitar un número y obtener su residuo o módulo

Leer numero

Si (numero mod 2) = 0 Entonces

*Indicar si es par o impar

Escribir = "Es par"

SiNo

Escribir "Es impar"

FinSi

*Cerrar el bucle cuando el iterador llegue a 10

Hasta que num = 10

Fin

Calculadora 'Al revés':

Inicio

***Inicializar variables, en inverso se almacenará el número invertido**

inverso = 0;

***Solicitar un número para invertirlo**

Escribir sin saltar "Ingrese un número";

Leer numero;

***Guardar el número ingresado en la variable residuo**

Residuo = numero;

***Realizar un bucle para invertir el número**

Mientras residuo > 0 Hacer

***Agregar un dígito a inverso y eliminar un dígito a residuo**

inverso = inverso * 10 + residuo mod 10;

residuo = (residuo – residuo mod 10) / 10;

Fin Mientras

***Mostrar el número invertido**

Escribir "valor del número inverso ", inverso;

Escribir "el número ", numero;

Fin

Conclusión

Estas tres calculadoras siguen unas secuencias lógicas de pasos que fueron diseñados para resolver problemas específicos de una manera eficiente. En los tres casos se ejemplifica como se podrían aplicar estructuras básicas de programación tales como los bucles, los condicionales y los operadores matemáticos. Por lo que ahora, con esta información, ya podríamos representar los algoritmos mediante el uso de diagramas de flujo y posteriormente escribir en algún lenguaje de programación a estas calculadoras. Considero también que el objetivo de la actividad se cumplió, ya que pude entender como funcionaba cada programa. Se me habría complicado más entenderlo si solo hubiera visto el código. De igual forma, me parece que sería adecuado en mis siguientes actividades realizar los algoritmos antes de escribir el código, incluso si no me es solicitado, ya que podré comprenderlo de mejor manera y detectar algunos posibles errores. Como se menciono al inicio, los algoritmos son los componentes básicos y esenciales de la informática, que permiten la mejora continua de nuestras habilidades como desarrolladores.

Referencias

- Gómez, L. (s. f.). Algoritmos. Universidad Autónoma del Estado de Hidalgo. <https://www.uaeh.edu.mx/scige/boletin/prepa4/n10/e1.html>
- ¿Qué es un algoritmo? (2018). NIC Argentina. <https://nic.ar/es/enterate/novedades/que-es-algoritmo>

Liga Github: <https://github.com/AlanGNB/IDS>