



APUNTES GUÍA VANKVERSITY SEMANA 1 PRINCIPAL

En esta guía encontraremos:

1. Explicación de que es un algoritmos y ejercicios
2. conceptos básicos de lógica con una tabla que incluye capturas de ejemplos
3. análisis de un algoritmo, un ejemplo de como resolver un problema
4. introducción a javascript
5. repositorios de git/github

1. QUÉ ES UN ALGORITMO

Definición

Un algoritmo es una serie de pasos lógicos ordenados y finitos para llegar a la solución de un problema, consta de tres partes:

Entrada de datos: son los datos que requiere el algoritmo para ser ejecutado

Proceso: es la secuencia de pasos lógicos que se llevan a cabo durante la ejecución del algoritmo

Salida: son los datos que se obtienen después de la ejecución del algoritmo

Ejercicios

Construya un algoritmo para:

- Preparar arroz
- Enviar un correo electrónico
- Cepillarse los dientes

Ejercicios resueltos

2. CONCEPTOS BÁSICOS DE LÓGICA

Ver la tabla con los conceptos básicos

3. ANÁLISIS DE UN PROBLEMA EN PROGRAMACIÓN

Pasos ordenados para resolver un problema en programación

1. Analizar el problema determinado, qué respuesta debe dar nuestro algoritmo a construir y cuál es el tipo de esta
2. (Entrada de datos) leer el problema a resolver y extraer las siguientes cosas:
 - Datos que requiere el algoritmo para trabajar
 - De los datos que requiere el algoritmo, ¿Cuáles nos los proporciona el enunciado?
 - De los datos que requiere nuestro algoritmo, ¿Cuáles NO los proporciona el enunciado del problema y debemos obtenerlos de otras fuentes?
3. (proceso) analizar cuáles son las acciones que debe realizar nuestro algoritmo, cuáles no los proporciona el enunciado del problema y debemos obtenerlos de otras fuentes
4. (salida) Construir la respuesta del algoritmo



Podemos usar el pseudocódigo para ayudarnos en este proceso en un principio

Ejemplo

Realizar un algoritmo para determinar la distancia de dos puntos en el espacio bidimensional

1. La respuesta debe ser numérica, ya que nos pide la distancia
2. **Entrada de datos:**

Ya que habla de distancia entre dos puntos en un espacio bidimensional, ya sabemos que necesitaremos las coordenadas de dos puntos en el espacio, ya que el enunciado no proporciona estas coordenadas, significa que se las pediremos al usuario

variables (todas ingresadas por el usuario)

coordenada_x1

coordenada_y1

coordenada_x2

coordenada_y2

3. Proceso:

¿Qué se debe hacer?

Pedir al usuario las coordenadas de los puntos

Calcular la distancia entre las coordenadas obtenidas

¿Cómo se hace lo que se debe hacer?

Pedir las coordenadas	usamos el prompt() para pedir los datos, con parseFloat() los convertimos a número para poder operar sobre ellos
Calcular la distancia	Usamos la formula de distancia entre dos puntos: para usar la raiz cuadrada usamos la librería Math con la función sqrt, para elevar números usamos **

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

formula de distancia entre dos puntos

4. Salida de datos:

¿Qué se debe hacer?

Imprimir un mensaje al usuario mostrando la distancia entre los puntos

¿Cómo se hace lo que se debe hacer?

Usando console.log()

Pasar el ejemplo a javascript

1. Entrada de datos

Las ingresa el usuario usando el prompt() y se convierten a float

```
/*ENTRADA DE DATOS*/  
  
coordenada_x1 = parseFloat(prompt("Ingrese la coordenada x1"));  
coordenada_x2 = parseFloat(prompt("Ingrese la coordenada x2"));  
coordenada_y1 = parseFloat(prompt("Ingrese la coordenada y1"));  
coordenada_y2 = parseFloat(prompt("Ingrese la coordenada y2"));
```

2. Proceso

Usamos Math.sqrt para aplicar la raiz, y ** para los exponentes

```
/*PROCESO*/  
  
distancia = Math.sqrt((coordenada_x2 - coordenada_x1)**2 + (coordenada_y2 - coordenada_y1)**2);
```

3. Salida de datos

Usamos console.log() para mostrar al usuario

```
/*SALIDA DE DATOS*/  
  
console.log("La distancia entre los puntos es: " + distancia);
```

Ejercicios

1. Realice un programa que calcule el índice de masa corporal de una persona.
2. Realice un programa que calcule el área de un trapecio isósceles.
3. Realice un programa que calcule la distancia entre dos puntos en el espacio tridimensional.

Ejercicios resueltos

JS SOLUCIÓN EJERCICIOS ANÁLISIS DE UN PROBLEMA EN PROGRAMACIÓN

4. INTRODUCCIÓN A JAVASCRIPT

Es el lenguaje más usado en el mundo, se usa tanto para front-end como para back-end, con js se pueden hacer apps móviles, servidores webs, videojuegos,

apps de escritorio, automatizaciones etc

Con js hay mercado laboral asegurado, su curva de aprendizaje es suave,

Por ahora lo estaremos ejecutando en el navegador

Características principales

- Es un lenguaje interpretado y se ejecuta en tiempo real
- Es dinámico ya que sus tipos de datos se asignan automáticamente
- Es orientado a eventos
- Es multiplataforma
- Es versátil

Entorno de desarrollo

- Navegador → Es el entorno más común, viene con una consola para probar y depurar el código
- Node.js → Es un entorno de ejecución que usa JavaScript como back-end, permite desarrollar fuera del navegador
- IDEs → se pueden usar editores de código como vscode o sublime text

4. REPOSITORIOS DE GIT/GITHUB

Git y GitHub son herramientas fundamentales para el control de versiones y la colaboración en proyectos de desarrollo de software.

Git:

- Sistema de control de versiones distribuido
- Permite rastrear cambios en el código fuente
- Facilita la colaboración entre desarrolladores
- Funciona localmente en tu máquina

Ventajas de Git:

- Control detallado del historial de cambios
- Trabajo offline

- Branching y merging eficiente

Podemos ver una lista grande de comandos de Git en la guía

GitHub:

- Plataforma basada en la nube que utiliza Git
- Aloja repositorios de código
- Proporciona herramientas adicionales para colaboración y gestión de proyectos

Ventajas de GitHub:

- Facilita la colaboración global
- Proporciona visibilidad y descubrimiento de proyectos
- Ofrece herramientas como issues, pull requests y acciones de CI/CD

Uso y diferencias entre Git y GitHub:

- Git se usa principalmente a través de la línea de comandos o GUI locales
- GitHub se usa a través de su interfaz web o aplicaciones de escritorio
- Git maneja el control de versiones, GitHub añade características sociales y de colaboración

Enlace del repositorio donde estoy almacenando todo lo del semillero

<https://github.com/valaristizabal/SEMILLERO-VANK-VERSITY-JS>