**Guia de Estudio / Cuestionario**

1. **¿Qué es un Sistema Informático (SI)?**

Conjunto de elementos físicos o hardware que son necesarios para la explotación de las aplicaciones informáticas o software.

El sistema informático o hardware es tangible, es decir, se puede ver y tocar (monitor, teclado,

procesador, memoria). Los programas o aplicaciones informáticas, así como el propio sistema

operativo, son intangibles; son software, pero no se puede tocar ni ver el conjunto de

instrucciones del que están formados.

Entre software y hardware existe otro concepto importante dentro de un sistema informático: el firmware. Es la parte intangible (software) de componentes del hardware. Es el caso del software con el que están programadas las memorias ROM, que son hardware. Sirva de ejemplo el software pregrabado que incorporan los teléfonos móviles para realizar todas las funciones a las que están destinados. El firmware no es fácilmente modificable. Una vez que se introduce o se graba en un componente hardware, queda prácticamente invariable a lo largo de la vida del ordenador. El firmware es, por tanto, software introducido en componentes electrónicos o hardware.

1. **Explique idea acerca del concepto de "PROBLEMA"**

Un problema es un asunto o cuestión que se debe solucionar o aclarar, una contradicción o un conflicto entre lo que es y lo que debe ser, una dificultad o un inconveniente para la consecución de un fin o un disgusto, una molestia o una preocupación.

1. **¿Qué cree que significa "PROGRAMAR"?**

La programación **informática** se refiere a la creación de software y el cómo se le indica al dispositivo lo **que** tiene **que** hacer mediante instrucciones (**programar**). **Informática** es el **qué** y la programación **informática** es el cómo. La programación resuelve problemas y crea herramientas.

1. **Cuáles son los pasos que realizaría para resolver un problema**

La solución de un problema por computadora, requiere de siete pasos, dispuestos de tal forma que cada uno es dependiente de los anteriores, lo cual indica que se trata de un proceso complementario y por lo tanto cada paso exige el mismo cuidado en su elaboración. Los siete pasos de la metodología son los siguientes:

1. **Definición de problema:** Es el enunciado del problema, el cual debe ser claro y completo. Es fundamental conocer y delimitar por completo el problema, saber qué es lo que se desea que realice la computadora, mientras esto no se conozca del todo, no tiene caso continuar con el siguiente paso.
2. **Análisis de la solución:** Consiste en establecer una serie de preguntas acerca de lo que establece el problema, para poder determinar si se cuenta con los elementos suficientes para llevar a cabo la solución del mismo.
3. **Diseño de la solución:** Una vez definido y analizado el problema, se produce la creación del algoritmo (Diagrama de flujo o pseudocódigo) en el cual se da la serie de pasos ordenados que nos proporcione un método explícito para la solución del problema.
4. **Codificación:** Consiste en escribir la solución del problema (de acuerdo al pseudocódigo); en una serie de instrucciones detalladas en un código reconocible por la computadora; es decir un lenguaje de programación (ya sea de bajo o alto nivel), a esta serie de instrucciones se le conoce como PROGRAMA.
5. **Prueba y depuración:** Prueba es el proceso de identificar los errores que se presenten durante la ejecución de programa, es conveniente que cuando se pruebe un programa se tomen en cuenta los siguientes puntos:
   1. Trata de iniciar la prueba con una mentalidad saboteadora, casi disfrutando la tarea de encontrar un error.
   2. Sospechar de todos los resultados que arroje la solución, con lo cual se deberán verificar todos.
   3. Considerar todas las situaciones posibles, normales y aun las anormales.

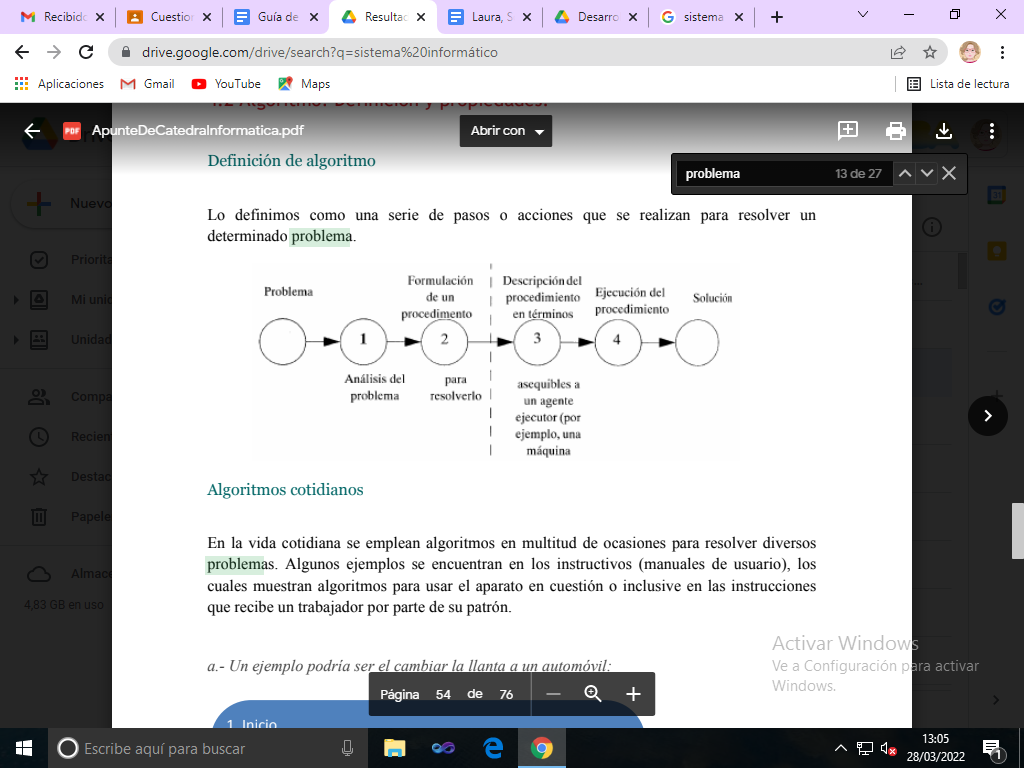
La depuración consiste en eliminar los errores que se hayan detectado durante la prueba, para dar paso a una situación adecuada y sin errores.

1. **Documentación**: Es la guía o comunicación escrita que sirve como ayuda para usar el programa, o facilitar futuras modificaciones. A menudo, un programa escrito por una persona es usado por muchas otras, por ello la documentación es muy importante; esta debe presentarse en tres formas: EXTERNA, INTERNA y al USUARIO FINAL.
   1. **Interna**: Consiste en los comentarios o mensajes que se agregan al código de programa, que aplican las funciones que realizan ciertos procesos, cálculos o fórmulas, para el entendimiento del mismo.
   2. **Externa**: Está integrada por los siguientes elementos: Descripción del problema, nombre del autor, diagrama de flujo y/o pseudocódigo, listas de variables y constantes, y codificación del programa, esto con la finalidad de permitir su posterior adecuación a los cambios.
   3. **Usuario Final**: es la documentación que se le proporciona al usuario final, es una guía que indica al usuario como navegar en el programa, presentando todas las pantallas y menús que se va a encontrar y una explicación de los mismos, no contiene información de tipo técnico.
2. **Mantenimiento**: Se lleva a cabo después de terminado el programa, cuando se ha estado trabajando un tiempo, y se detecta que es necesario hacer un cambio, ajuste y/o complementación al programa para que siga trabajando de manera correcta. Para realizar esta función el programa debe estar debidamente documentado, lo cual facilitará la tarea.
3. **¿Cuáles serían las competencias de un programador?**

* Dominar varios lenguajes de programación
* Tener conocimientos sobre BD
* Capacidad para analizar y resolver problemas
* Tener un buen dominio del inglés
* Poseer conocimientos técnicos sobre informática
* Poseer capacidad para organizar su trabajo
* Dominar técnicas autodidactas
* Habilidad para el trabajo individual y en equipo
* Capacidad para manejar altos volúmenes de trabajo.

1. **Explique el concepto algoritmo y de una definición para dicho concepto y describa alguno**

serie de pasos o acciones que se realizan para resolver un determinado problema.



**Algoritmos cotidianos**

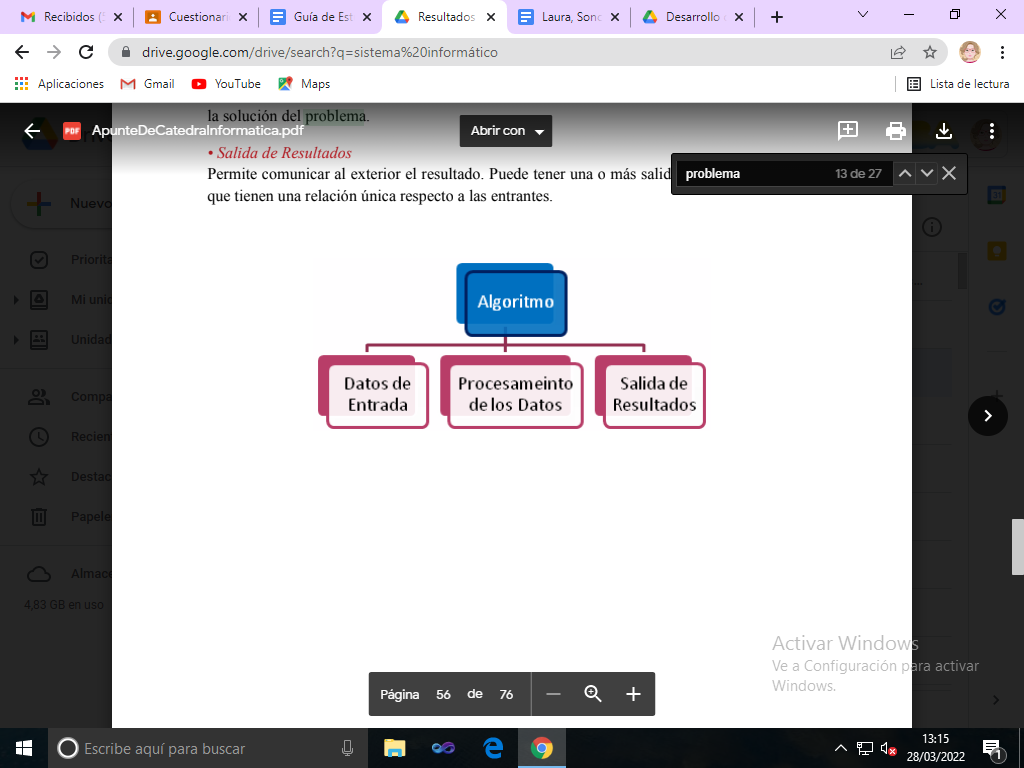
En la vida cotidiana se emplean algoritmos en multitud de ocasiones para resolver diversos problemas. Algunos ejemplos se encuentran en los instructivos (manuales de usuario), los cuales muestran algoritmos para usar el aparato en cuestión o inclusive en las instrucciones que recibe un trabajador por parte de su patrón.

Un ejemplo podría ser el cambiar la llanta a un automóvil:

1. Inicio.
2. Traer crique o gato hidráulico.
3. Levantar el coche con el crique.
4. Aflojar tornillos de las llantas.
5. Sacar los tornillos de las llantas.
6. Quitar la llanta.
7. Poner la llanta de repuesto.
8. Poner los tornillos.
9. Apretar los tornillos.
10. Bajar el crique o gato hidráulico
11. Fin

Ejemplo de un algoritmo para preparar una taza de café, tomando en cuenta que se tiene agua caliente, una taza, cuchara, café, crema y azúcar.

1. Inicio.
2. Verter agua caliente en la taza.
3. Tomar con la cuchara el café.
4. Poner el café en la taza.
5. ¿Quieres azúcar?

Si 6 No 8

1. Tomar el azúcar con la cuchara
2. Poner el azúcar en la taza.
3. ¿Quieres Crema?

Si 9 No 11

1. Tomar la crema con la cuchara.
2. Poner crema en la taza
3. Revolver.
4. Fin
5. **¿Cuál cree que es la función de un lenguaje?**

El **lenguaje** es una función superior que desarrolla los procesos de simbolización relativos a la codificación y decodificación. La producción del **lenguaje** consiste en la materialización de signos (sonoros o escritos) que simbolizan objetos, ideas, etc, de acuerdo con una convención propia de una comunidad lingüística.

Está compuesto por un sistema de signos que utiliza una comunidad para comunicarse oralmente o por escrito. Por ej. Los seres humanos tenemos una capacidad propia para expresar pensamientos y sentimientos por medio de la palabra.

1. **¿Y la función de un lenguaje de programación?**

Un **lenguaje de programación** es una forma de comunicarnos con una computadora, tablet o celular e indicarle qué queremos hacer. Existen distintos tipos de **lenguaje**: principalmente de bajo nivel y de alto nivel. La diferencia se encuentra en lo cerca o lejos que estemos del hardware de nuestro equipo.

El pseudocódigo es un lenguaje de especificación (descripción) de algoritmos. El uso de tal lenguaje hace el paso de codificación final (esto es, la traducción a un lenguaje de programación)relativamente fácil.

El pseudocódigo nació como un lenguaje similar al inglés. Se considera un primer borrador, dado que el pseudocódigo tiene que traducirse posteriormente a un lenguaje de programación.

El pseudocódigo no puede ser ejecutado por una computadora. La ventaja del pseudocódigo es que en su uso, en la planificación de un programa,el programador se puede concentrar en la lógica y en las estructuras de control y no preocuparse de las reglas de un lenguaje específico. Es también fácil modificar el pseudocódigo si se descubren errores o anomalías en la lógica del programa, mientras que en muchas ocasiones suele ser difícil el cambio en la lógica, una vez que está codificado en un lenguaje de programación.

1. **¿ Qué lenguajes de programación conoce?**

C++, VB .NET, Java, HTML, Python, JavaScript, SQL, Assembler

1. **¿Conoce qué tipos de datos maneja un lenguaje computacional?**

Los **datos** son representaciones simbólicas (vale decir: **numéricas**, **alfabéticas**, **algorítmicas**, etc.) de un determinado **atributo** o **variable cualitativa** o **cuantitativa**, o sea: la descripción codificada de un hecho empírico, un suceso, una entidad.

**Los tipos de datos** comunes son: números enteros, números con signo (negativos), números de coma flotante (decimales), cadenas alfanuméricas, estados (boolean), char (carácter), binario, BCD (Binario codificado a decimal), hexadecimal, ASCII, etc.

1. **¿ Tiene conocimiento sobre tipos de estructuras de datos que utilizan los lenguajes computacionales? ¿Cuáles?**

En programación, una estructura de datos es una forma de organizar un conjunto de datos elementales con el objetivo de facilitar su manipulación. Un dato elemental es la mínima información que se tiene en un sistema.

Una **estructura de datos** define la organización e interrelación de éstos y un conjunto de operaciones que se pueden realizar sobre ellos. Las **operaciones básicas** son:

**Alta**, adicionar un nuevo valor a la estructura.  
**Baja**, borrar un valor de la estructura.  
**Búsqueda**, encontrar un determinado valor en la estructura para realizar una operación con este valor, en forma SECUENCIAL o BINARIO (siempre y cuando los datos estén ordenados)…  
**Otras operaciones** que se pueden realizar son:

**Ordenamiento**, de los elementos pertenecientes a la estructura.  
**Apareo**, dadas dos estructuras originar una nueva ordenada y que contenga a las apareadas.

Cada estructura ofrece ventajas y desventajas en relación a la simplicidad y eficiencia para la realización de cada operación. De esta forma, la elección de la estructura de datos apropiada para cada problema depende de factores como la frecuencia y el orden en que se realiza cada operación sobre los datos.

**Una estructura de datos es una clase de datos que se puede caracterizar por su organización y operaciones definidas sobre ella.**