



# Unidad 1 - R

*Duarte Juan Ignacio*

## VERDADERO y FALSO

1. Un algoritmo es un conjunto infinito de operaciones que permiten resolver un tipo de problema.
2. El concepto de algoritmo solo se aplica a problemas que serán resueltos por computadoras.
3. Un algoritmo debe ser aplicable a un problema específico y no a un tipo de problemas.
4. Las recetas de cocina son ejemplos de algoritmos.
5. Un algoritmo siempre debe ser enunciado en un lenguaje de programación.
6. La eficacia de un algoritmo no depende de los recursos con los que se cuente.
7. Para cualquier problema que pueda especificarse un método finito de solución, puede definirse un algoritmo.
8. Un ejecutante es una entidad capaz de entender y ejecutar los pasos descritos en un algoritmo.
9. Una acción no-primitiva no puede ser descompuesta en acciones primitivas.
10. La programación modular es un método de diseño que divide un problema en partes independientes.
11. La programación modular disminuye la complejidad del problema original.
12. Un módulo en la programación modular debe tener múltiples entradas y salidas.
13. La programación modular facilita las modificaciones y conexiones en la resolución de un problema.

# PREGUNTAS y RESPUESTAS

1:

**¿Cuáles son las etapas principales para la resolución de problemas computacionales?**

R: Las etapas principales para la resolución de problemas computacionales son: Definición, Análisis, Programación y Prueba.

2:

**¿Qué tres conjuntos se deben identificar durante el análisis de un problema computacional?**

R: Durante el análisis de un problema computacional, se deben identificar los conjuntos de Datos, Resultados y Condiciones.

3:

**¿Cuáles son las dos subetapas que componen la etapa de programación en la resolución de problemas computacionales?**

R: Las dos subetapas que componen la etapa de programación en la resolución de problemas computacionales son la de elección y creación del método y codificación.

4:

**¿Qué es una estrategia en el contexto de la resolución de problemas computacionales y cómo se diseña?**

R: Una estrategia en el contexto de la resolución de problemas computacionales es un plan que indica las tareas a realizar para resolver el problema. Se diseña mediante el análisis detallado del problema y la identificación de las operaciones necesarias para transformar las entradas en las salidas deseadas.

5:

**¿Qué es el diseño descendente (top-down) y cuáles son sus características generales?**

R: El diseño descendente (top-down) es un enfoque en el que se inicia con el problema general y se va descomponiendo en subproblemas más pequeños y manejables. Sus características generales incluyen la división de tareas complejas en tareas más simples y la resolución de problemas de manera sistemática.

6:

**¿Por qué es importante la definición correcta y completa del problema en la resolución de problemas computacionales?**

R: La definición correcta y completa del problema en la resolución de problemas computacionales es importante porque proporciona una comprensión clara del problema, lo que permite desarrollar una solución adecuada y eficaz.

7:

**¿Cuál es el objetivo de la etapa de prueba en la resolución de problemas computacionales?**

R: El objetivo de la etapa de prueba en la resolución de problemas computacionales es verificar que la solución desarrollada funcione como se espera y produzca los resultados deseados.

8:

**¿Qué se entiende por "mejor alternativa" al elegir un método para resolver un problema computacional?**

R: Al elegir un método para resolver un problema computacional, "mejor alternativa" se refiere a la opción que resuelve el problema de manera más eficaz y eficiente, considerando aspectos como la simplicidad, la eficiencia en términos de tiempo y espacio, y la adaptabilidad a cambios futuros.

9:

**¿En qué consiste la codificación durante la etapa de programación en la resolución de problemas computacionales?**

R: La codificación durante la etapa de programación en la resolución de problemas computacionales consiste en la implementación del algoritmo diseñado en un lenguaje de programación específico.

10:

**¿Cuál es el propósito de dividir un problema en subproblemas más simples en el diseño de una estrategia?**

R: El propósito de dividir un problema en subproblemas más simples en el diseño de una estrategia es hacer el problema más manejable y fácil de resolver, permitiendo abordar cada parte del problema individualmente antes de combinar las soluciones para obtener la solución completa.

11:

**¿Qué es un algoritmo y cuáles son sus características principales?**

R: Un algoritmo es un conjunto finito de operaciones que, seguidas en un determinado orden, permiten resolver un tipo de problema. Sus características principales son ser finito, definido y general.

12:

**¿Qué es un "ejecutante" en el contexto de la resolución de problemas computacionales?**

R: Un "ejecutante" es una entidad capaz de entender un enunciado y ejecutar los pasos descritos en un algoritmo.

13:

**¿Cuál es la diferencia entre una acción primitiva y una acción no-primitiva en un algoritmo?**

R: Una acción es primitiva cuando su enunciado es suficiente para ser ejecutada sin información adicional. Una acción no-primitiva puede ser descompuesta en acciones primitivas para un ejecutante dado.

14:

**¿Qué es una condición en el contexto de un algoritmo?**

R: Una condición es una afirmación lógica sobre el estado de algún recurso del ambiente que puede tomar valor verdadero o falso en el momento de la observación.

15:

**¿Qué es la programación modular y cuáles son sus objetivos?**

R: La programación modular es un método de diseño que divide el problema en partes independientes. Sus objetivos son disminuir la complejidad, aumentar la claridad, aumentar la confiabilidad y facilitar modificaciones y conexiones.

16:

**Mencione un ejemplo de un algoritmo en la vida diaria.**

R: Un ejemplo de un algoritmo en la vida diaria puede ser una receta de cocina.

17:

**¿La eficacia de un algoritmo depende de los recursos y factores considerados?  
¿Por qué?**

R: Si, la eficacia de un algoritmo depende de los recursos y factores considerados porque estos pueden influir en el costo y el tiempo de ejecución del algoritmo.

18:

**¿Qué es el ambiente del problema en el contexto de la resolución de problemas computacionales?**

R: El ambiente del problema es el conjunto de recursos existentes en el momento de la ejecución de un trabajo.

19:

**¿Cuál es el propósito de dividir un problema en módulos en la programación modular?**

R: Dividir un problema en módulos facilita su análisis, resolución y puesta a punto, al poder tratar cada parte de manera individual.

20:

**¿Qué criterios se deben tener en cuenta para determinar un módulo en la programación modular?**

R: Para determinar un módulo en la programación modular se deben tener en cuenta criterios como que el módulo corresponda a una función lógica bien definida, que sea de poca complejidad, que tenga una estructura de caja negra y que tenga una única entrada y salida.