



COMPONENTE FORMATIVO

NÚMERO DEL COMPONENTE FORMATIVO	1
NOMBRE DEL COMPONENTE FORMATIVO	Introducción a la algoritmia

COMPETENCIA	Desarrollar la solución de software de acuerdo con el diseño y metodologías de desarrollo.	RESULTADOS DE APRENDIZAJE	Implementar algoritmos en la resolución de procesos lógicos utilizando lenguaje de programación seleccionado.
-------------	--	---------------------------	---

Introducción

En la vida cotidiana se debe imaginar, diseñar y planear soluciones a problemas corrientes del día a día, por ende, muchas veces la estrategia de “divide y vencerás” es empleada para afrontar la solución de estos problemas la cual consiste en tomar un problema grande y segmentarlo en pequeños subproblemas (o pequeñas metas) que son más fáciles de abordar, y la unión logra la solución a lo inicialmente planteado.

Desde un punto de vista computacional esta estrategia es la más empleada y los grandes sistemas informáticos son la unión de un conjunto de instrucciones computacionales que resuelven problemas muy puntuales.

Desde aquí se propone facilitar técnicas, para el análisis de problemas usando algoritmos que a futuro pueden llegar a ser parte constitutiva de un sistema teleinformático.

1. Introducción a la algoritmia

La aplicación apropiada de una serie de pasos muy detallados, puede garantizar una solución correcta de un problema. La programación de computadores (o construcción de software) está basada en esta práctica.

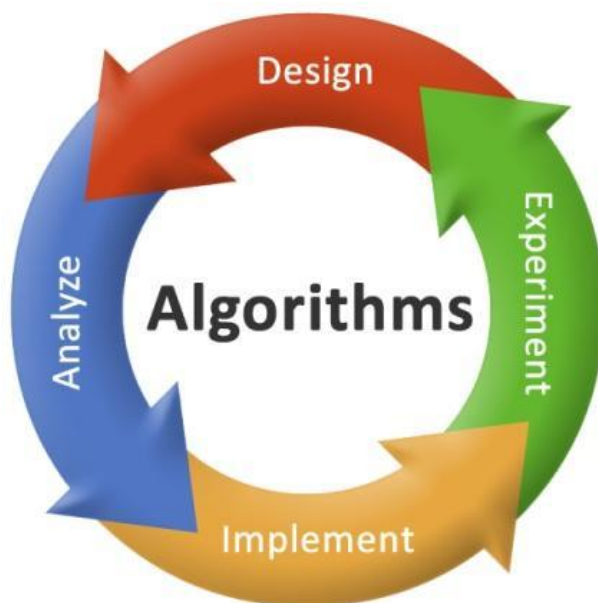


Imagen: pasos en la algoritmia básica.

1.1. Definición de algoritmo

Según la Real academia de la lengua española (RAE) un algoritmo está definido como: conjunto ordenado y finito de operaciones que permite hallar la solución de un problema (Asale, R.-, & Rae, 2021).

Entendiendo que computacionalmente las máquinas están en capacidad de hacer unas pocas operaciones matemáticas muy básicas y sencillas, pero también se considera que computan enormes cantidades de estas operaciones en unidades tiempo muy corto.

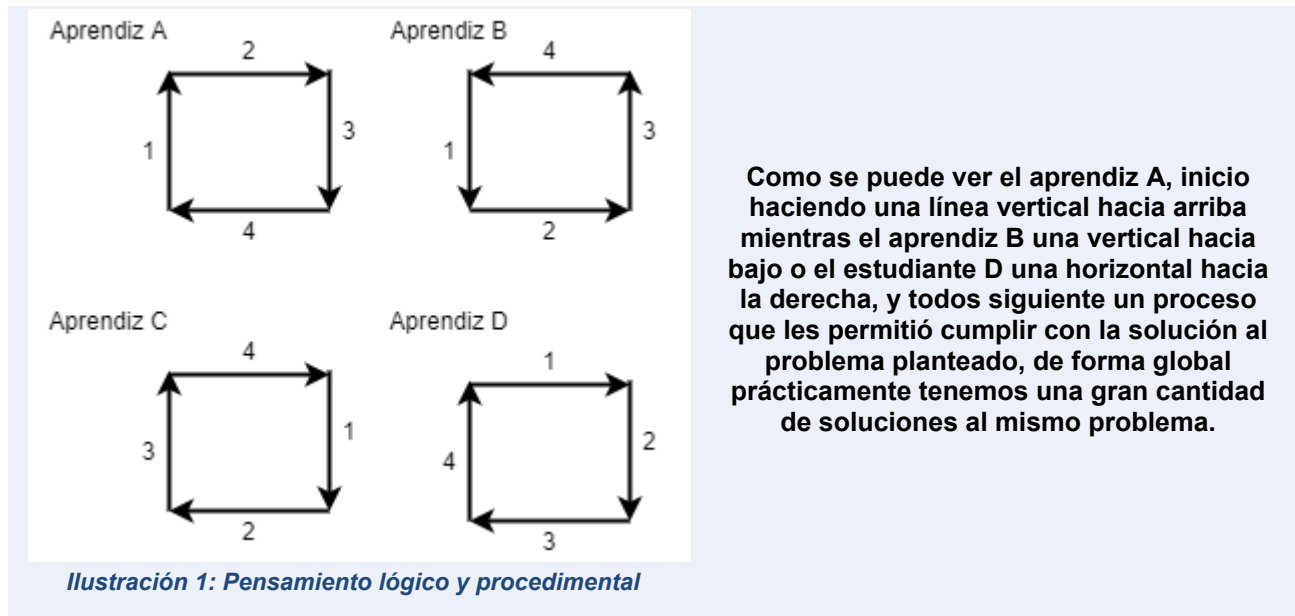
La solución de un problema usando computadoras fácilmente se puede convertir en una enorme cantidad de pequeñas operaciones matemáticas, tal vez muy abrumadoras desde el punto de vista humano. Es por eso que se requiere, la especialización de conjuntos de operaciones que se agrupan en pequeños procesos que componen la solución de un problema.

Para ampliar la definición puede visualizar el siguiente vídeo:
<https://www.youtube.com/watch?v=U3CGMyjzlvM>

1.2. Pensamiento algorítmico

Un problema puede ser abordado de varias maneras y aportar la misma solución, pero lo que siempre tendrán en común es que para resolverlo usaron una secuencia de pasos ordenados.

Por ejemplo: si se pasa una hoja de papel cuadriculado a los aprendices de una clase y les pide que dibujen un cuadrado sin levantar el lápiz habrá quienes lo hagan de diferente manera y obtengan el mismo resultado cómo muestra la siguiente imagen:



Imaginando que el problema es mucho más complejo que el descrito en el ejemplo se concluye que existen tantas soluciones como el enfoque o lógica empleada por cada aprendiz o persona.

Pero lo que sí es común en todos los casos es que se realizaron unas fases mentales secuencialmente ordenadas que son:

1. Entender el problema enunciado por el instructor. (Entender el problema)
2. Escoger una ruta a seguir del rectángulo (Hacer un plan)
3. Plasmar en el papel cuadriculado ese plan. (Ejecutar el plan)
4. Revisar el resultado y ver que está completado el objetivo. (Revisar)

Cuando el problema es muy complejo de resolver, muchas veces toca volver y analizar el problema o trazar un nuevo plan o plasmar ese nuevo plan una y otra vez revisando los resultados y volver a analizar hasta encontrar la solución al problema.

1.3. Solución de problemas y programación

En el ámbito de la computación, los algoritmos son una herramienta que permite describir claramente un conjunto finito de instrucciones, ordenadas secuencialmente y libres de ambigüedad, que debe llevar a cabo un computador para lograr un resultado previsible. Vale la pena recordar que un programa de computador consiste de una serie de instrucciones muy precisas y escritas en un lenguaje de programación que el computador entiende (Logo, Java, Pascal, etc). (López, 2019)

Ejemplo: se propone diseñar un algoritmo para apagar una computadora.



```
Algoritmo: apagar computadora
INICIO
1.-Dar clic en el icono INICIO
2.-Dar clic en el icono que dice APAGAR
3.-Esperar que se apague completamente
4.-Desconectar los cables del enchufe
FIN
```

Ejemplo: se propone diseñar un algoritmo para cargar un celular

```
Algoritmo: Cargar un celular
INICIO.
1. Buscar el cargador.
2. Agarrar el cargador.
3. Agarrar el celular.
4. Poner el enchufe en el celular
5. Conectarlo a tomacorriente.
6. Esperar a que se ponga a cargar.
7. Desconectarlo del tomacorriente.
8. Ponerlo en un lugar seguro.
FIN
```

Con base a los ejemplos anteriormente planteados se puede concluir, que un algoritmo debe ser:

- ✓ **Realizable:** el proceso algorítmico debe terminar después de una cantidad finita de pasos. Se dice que un algoritmo es inaplicable cuando se ejecuta con un conjunto de datos iniciales y el proceso resulta infinito o durante la ejecución se encuentra con un obstáculo insuperable sin arrojar un resultado (Lopez, 2009).
- ✓ **Comprensible:** debe ser claro lo que hace, de forma que quien ejecute los pasos (ser humano o máquina) sepa qué, cómo y cuándo hacerlo. Debe existir un procedimiento que determine el proceso de ejecución (Lopez, 2009).
- ✓ **Preciso:** el orden de ejecución de las instrucciones debe estar perfectamente indicado. Cuando se ejecuta varias veces, con los mismos datos iniciales, el resultado debe ser el mismo siempre. La precisión implica determinismo (Lopez, 2009).

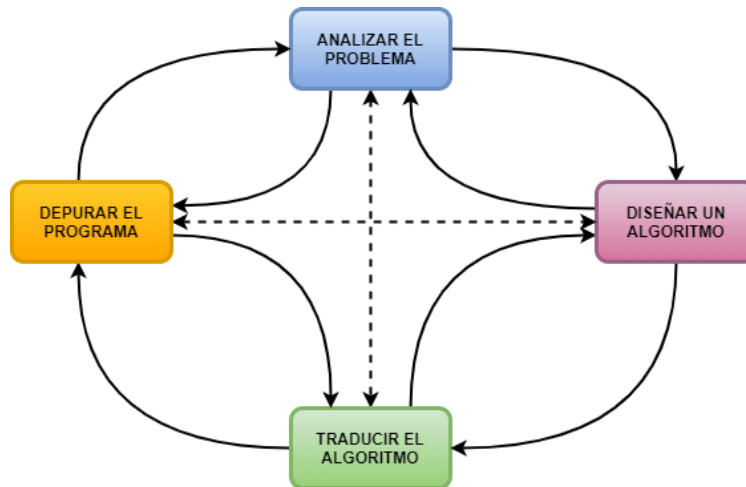


Ilustración 2: Fases del ciclo de programación

La ilustración 2 muestra las fases y sus relaciones en el proceso de diseño de algoritmos de programas informáticos, que están basados en un algoritmo que posteriormente es escrito en un lenguaje de programación.

1.4. Análisis del problema (entenderlo)

Cómo los programas de computador tienen como objetivo resolver problemas muy puntuales lo primero que se debe realizar para resolverlas es lograr una mejor comprensión posible de estas.

La forma de realizar esta actividad consiste en:

- I. Formular claramente el problema
- II. Especificar los resultados que se desean obtener con la solución.
- III. Identificar la información disponible los datos.

1.4.1. Formular el problema

Algunas preguntas modeladoras antes de formular el problema son:

- ✓ ¿Puedo definir de forma más clara el problema?
- ✓ ¿Qué palabras no me son conocidas o desconozco el significado?
- ✓ ¿He resuelto antes algún problema similar?
- ✓ ¿Qué información es importante para resolver el problema?
- ✓ ¿Qué información NO es importante y puedo omitir?

Ejemplo



Don Juan necesita decidir cómo comprar un teléfono celular que cuesta \$870.000 de contado o \$980.000 a crédito, él tiene \$630.000 pesos en efectivo.

Luego de revisar las preguntas modeladoras se puede concluir:

Como el efectivo que tiene don Juan no le alcanza para comprar el teléfono celular de contado, entonces él tiene dos opciones: comprarlo completamente a crédito o también pagar una parte al contado (como cuota inicial) y el resto diferirlo a crédito.

Para poder resolver este problema se debe conocer el número de cuotas en la que difiere si desea pagarlo totalmente a crédito o conocer el número de cuotas y el valor total del celular si se da una cuota inicial de \$630.000 pesos.

1.4.2. Precisar elementos de entrada

Otro punto importante en la fase de análisis del problema es determinar cuál es la información disponible. Por lo tanto:

- ✓ ¿Qué información es importante y necesaria para resolver el problema?
- ✓ ¿Qué información no es importante y se puede prescindir?
- ✓ ¿Cuáles son los datos de entrada conocidos?
- ✓ ¿Cuál es la incógnita o que debemos calcular?
- ✓ ¿Los datos se pueden agrupar en categorías?
- ✓ ¿Qué información adicional hace falta para resolver el problema?

Otro elemento a resaltar es el nivel de conocimiento y las limitaciones que se tienen en el ámbito o el contexto del problema que está tratando de resolver. Por ejemplo, si el problema requiere conceptos financieros para entender la naturaleza de los datos, o si es necesario adquirir nueva información y/o poder consultarla, para estar seguros de que es posible tratarla conforme el contexto lo requiera.

1.4.3. Precisar resultados esperados

Determinar con claridad cuál es el resultado final esperado (o producto) que se quiere obtener, es necesario determinar o definir qué resultados se solicitan y en qué tipo, o formato deben tener esos (mostrados en pantalla, diagramación, orden, etc.). Para lograrlo es bueno preguntarse:

- ✓ ¿Qué información están solicitando?
- ✓ ¿En qué formato se debe entregar esta información?

1.5. Desarrollo de la creatividad elementos, modelos, fases y sus objetivos.



Resolver algoritmos es una actividad que requiere en la mayoría de los casos creatividad por parte del programador, esta es una habilidad que se va fortaleciendo en la medida que se desempeña más y más en procesos de desarrollo de software.

Para estimular el pensamiento creativo propuesto por (Resnick, 2007) el programador debe imaginar que quiere hacer; crea un proyecto basado en sus ideas; juega y comparte sus ideas y creaciones con otros para reflexionar sobre sus experiencias y permitirse imaginar nuevas ideas y nuevos proyectos. La espiral genera un proceso indefinido de mejoramiento continuo (López, 2009).



Ilustración 3. Espiral del Pensamiento Creativo diseñada por el Dr. Mitchel Resnick

En conclusión, se puede decir que un algoritmo es la definición de una secuencia lógica de pasos que permiten solucionar un problema. Formular algoritmos es la base de los grandes adelantos en las ciencias matemáticas y tecnología, ya que se pueden aplicar para cualquier área del conocimiento.

2. Lógica matemática

La lógica matemática es la disciplina que trata de métodos de razonamiento. En un nivel elemental, la lógica proporciona reglas y técnicas para determinar si es o no válido un argumento dado. El razonamiento lógico se emplea en matemáticas para demostrar teoremas; en ciencias de la computación para verificar si son o no correctos los programas; en las ciencias físicas y naturales, para sacar conclusiones de experimentos; y en las ciencias sociales y en la vida cotidiana, para resolver una multitud de problemas. Ciertamente se usa en forma constante el razonamiento lógico para realizar cualquier actividad (Jiménez 2007).



Para ampliar los fundamentos de la lógica matemática puede revisar el vídeo:
<https://www.youtube.com/watch?v=9zYDaZhS7Ac>

2.1. Componentes de la lógica matemática

Todo razonamiento lógico está compuesto por una serie de elementos, que se interrelacionan entre sí, la correcta interpretación de estos componentes y sus relaciones son lo que denominamos el análisis deductivo.

Estos componentes son las preposiciones y conectores lógicos descritos a continuación:

A. Proposiciones

Una proposición o enunciado es una oración que puede ser falsa o verdadera pero no ambas a la vez. La proposición es un elemento fundamental de la lógica matemática.

Las proposiciones pueden ser simples o compuesta, las primeras están compuestas por una sola proposición, por el contrario, en las proposiciones compuestas están conformadas por dos o más proposiciones unidas por un conector.

A continuación, se muestran una serie de ejemplos de proposiciones válidas y no válidas, con estos se explicará por qué algunos enunciados no son proposiciones.

p: La tierra es plana

q: $-17 + 38 = 21$

r: Hola ¿Cómo estás?

S: Lava el Carro por favor

Para los enunciados anteriores podemos determinar lo siguiente:

p y q son proposiciones simples válidas porque pueden tomar un valor verdadero o falso, sin embargo, los enunciados r y s son proposiciones no válidas, ya que no pueden tomar un valor de falso o verdadero, uno de ellos es un saludo y el otro es una orden.

B. Conectores lógicos y proposiciones compuestas.

Existen las proposiciones compuestas que son las que se componen de varias proposiciones las cuales utilizan para su conexión operadores o conectores lógicos que son:

✓ Operador AND (y) Conjunción

Conecta dos proposiciones que se deben unir para que se pueda obtener un resultado verdadero, esta se representa por el siguiente símbolo \wedge

Ejemplos:



- Nuestra bandera es de color amarillo, azul y rojo
- El número 9 es mayor que 7 y menor que 12
- El carro enciende cuando tiene corriente eléctrica y gasolina en el tanque.

✓ Operador OR (o) Disyunción

Conecta dos proposiciones y se obtiene una proposición compuesta verdadera cuando por lo menos unas de las proposiciones son verdaderas.

Ejemplos:

- Puedo ir a Bogotá en avión o en bus
- Puedo entrar al estadio si compro mi boleta o si tengo un pase de cortesía
- Voy a desayunar con huevos revueltos o huevos pericos

✓ Operador NOT (no) Negación.

Su propósito es negar una proposición es decir que si tenemos una proposición verdadera al aplicarle este tipo de operador se obtiene la negación es decir falso.

Ejemplos:

- No es cierto que Gabriel García Márquez escribió la *Ilíada*.
- No es cierto que el América de Cali es de la ciudad de Bogotá.
- Es falso que el agua es un hidrocarburo

2.2. Tablas de verdad

Es una estrategia de la lógica que tiene como objetivo determinar la validez de varias propuestas en alguna situación, es decir, determina las condiciones necesarias en las que un enunciado es verdadero, estas condiciones se representan con una tabla de verdad que determina el valor de verdad de todas las proposiciones que componen la situación. Así como lo explica el principio de bivalencia: **una proposición puede ser verdadera o falsa únicamente, nunca ambas.**

Algunas consideraciones:

- El valor de una proposición verdadera se representa por "V", o un "1"
- El valor de una proposición falsa se representa por "F" o un "0"
- Existe una tabla de verdad para cada uno de los conectores lógicos, lo que se tiene que evaluar para responder si una proposición compuesta es verdadera o falsa es el estatus actual de las proposiciones simples. A continuación, se muestran las tablas de verdad.

P	Q	$P \wedge Q$
V	V	V
V	F	F
F	V	F
F	F	F

O puede ser

P	Q	$P \wedge Q$
1	1	1
1	0	0
0	1	0
0	0	0



Tabla 1. Tabla de verdad Conjunción(Y)

P	Q	$P \vee Q$
V	V	V
V	F	V
F	V	V
F	F	F

O puede ser

P	Q	$P \vee Q$
1	1	1
1	0	1
0	1	1
0	0	0

Tabla 2. Tabla de verdad Disyunción (O)

P	$\neg P$
V	F
F	V

O puede ser

P	$\neg P$
1	0
0	1

Tabla 3: Tabla de verdad Negación (NO)

Existe otro tipo de conectores que llamaremos proposiciones condicionales que son aquellas que están formadas por dos proposiciones simples o compuesta de p y q la cual se representa y lee de la siguiente manera:

$p \rightarrow q$ se lee “Si p entonces q”

Ejemplo 1:

Si no llueve se arruina la cosecha.

Este tipo de proposición se conoce como la condición “para que” de manera que para que no se arruine la cosecha se requiere que llueva.

p: Si no llueve

q: se arruina la cosecha

Al aplicar el operador $p \rightarrow q$ su tabla de verdad quedaría de la siguiente manera.

P	Q	$P \rightarrow Q$
V	V	V
V	F	F
F	V	V

O puede ser

P	Q	$P \rightarrow Q$
1	1	1
1	0	0
0	1	1



F	F	V
---	---	---

0	0	1
---	---	---

Tabla 4: Tabla de verdad Condicional (SI... ENTONCES)

De igual manera existe una proposición bicondicional que indica que una proposición es verdadera SI y SOLO SI las proposiciones simples o compuestas que forman la proposición son verdaderas o son falsas si y solo si ambas proposiciones son falsas, esta representa y se lee de la siguiente manera:

$p \leftrightarrow q$ se lee "p si solo si q"

Ejemplo 2:

Es buen estudiante, si y solo sí; tiene promedio de cinco, donde se puede concluir lo siguiente:

p: es un buen estudiante

q: tiene promedio de 5

al aplicar el operador $p \leftrightarrow q$ su tabla de verdad quedaría de la siguiente manera:

P	Q	$P \leftrightarrow Q$
V	V	V
V	F	F
F	V	F
F	F	V

O puede ser

P	Q	$P \leftrightarrow Q$
1	1	1
1	0	0
0	1	0
0	0	1

Cuando se aplican tablas de verdad se puede llegar a tres tipos de resultado que son:

- ✓ Tautología: es una expresión lógica que es verdadera para todos los posibles valores de verdad de sus componentes atómicos. Las proposiciones dan el resultado positivo (TOMi.digital, 2021).
- ✓ Contradicción: son aquellas fórmulas que son falsas para cualquier valoración de los símbolos proposicionales que contiene. Las proposiciones dan el resultado negativo (TOMi.digital, 2021).
- ✓ Contingencia: son aquellas fórmulas cuyo valor de verdad o falsedad depende de la valoración de los símbolos proposicionales que contiene. Las proposiciones dan un resultado negativo y positivo (TOMi.digital, 2021).

A continuación, se mostrará como aplicando lógica matemática podemos comprobar el valor de verdad de un enunciado, este tipo de práctica es muy útil en las computadoras, ya que permiten tener inferencia y deducir acciones o actividades a seguir.

Ejemplo 3:



P: todas las palomas vuelan

Q: todas las aves tienen plumas

R: luego todas las palomas son aves

Estas proposiciones se pueden escribir utilizando los operadores lógicos de la siguiente manera:

$$(p \wedge q) \rightarrow r$$

Para el anterior ejercicio se construye la tabla de verdad:

P	Q	R	$P \wedge Q$	$\rightarrow R$
V	V	V	V	V
V	V	F	V	F
V	F	V	F	V
V	F	F	F	V
F	V	V	F	V
F	V	F	F	V
F	F	V	F	V
F	F	F	F	V

2.3. Ejercicios de lógica proposicional

A continuación, se plantean a manera de ejemplo algunos ejercicios con su respectiva solución que permitirán identificar un problema, reconocer los datos de entrada y los procesos a ser aplicados como estrategia a seguir desde la utilización de la lógica proposicional en la identificación y solución de problemas:

2.3.1. Ejercicio 1:

Analizar el siguiente enunciado: “Se necesita personal femenino que sepa Microsoft Office o Gsuite.”
Para dar solución al enunciado anterior vamos a realizar las siguientes actividades.

Pasos a seguir:

- ✓ Identificar el problema
- ✓ Descomponer el enunciado en proposiciones atómicas
- ✓ Evaluar los posibles casos derivados del enunciado anterior
- ✓ Determinar los casos donde se cumpla la condición y hacer la selección de las personas de acuerdo al requerimiento.



Solución:

- ✓ Identificar el problema: se trata de buscar o contratar personal femenino que sepa utilizar Microsoft Office o Google Suite.
- ✓ Descomponer en proposiciones atómicas del enunciado (cuando están separadas del contexto del problema se llaman proposiciones atómicas).
 - p: es personal femenino
 - q: sabe utilizar el Microsoft office
 - r: sabe utilizar la suite de Google
- ✓ Determinar las relaciones lógicas de acuerdo a lo planteado en el enunciado utilizando las proposiciones atómicas definidas en el punto anterior
 - $p \wedge (q \vee r)$
- ✓ Evaluar los posibles casos derivados del enunciado anterior

2.3.2. Ejercicio 2:

Juan sabe utilizar Microsoft office y Gsuite, Martha sabe utilizar Gsuite, Natalia sabe utilizar Gsuite y Microsoft Office.

Crear la tabla de verdad y aplicar la lógica proposicional para elegir las personas que cumplan con los requisitos exigidos.

Para la construcción de la tabla de verdad vamos a evaluar cada una de las preposiciones atómicas e iremos validándolas para verificar si se cumple lo que la haría verdadera o si no se cumple lo cual haría que la proposición sea falsa.

Empezaremos a analizar cada posible caso por lo que evaluamos a Juan, para la primera proposición sería un valor falso, ya que p requiere que sea personal femenino el cual se pondrá en la columna 2, luego evaluamos si Juan sabe utilizar Microsoft Office o la Suite de Google el valor irá en la columna tres y por último evaluamos la conjunción para determinar la solución.

Nombre	p	$(q \vee r)$	$p \wedge (q \vee r)$



Juan	F	V	F
Martha	V	V	V
Natalia	V	V	V

De acuerdo a la anterior tabla podemos concluir que Martha y Natalia son actas para el trabajo lo que permite seleccionarlas.

2.3.3. Ejemplo 3.

Enunciado: “Sandra dice Adelgazo o no hago dieta”

Para este enunciado además se proponen una serie de alternativas que podrían ser validas o no dependiendo de la solución propuesta utilizando la lógica matemática, las siguientes son las posibles respuestas de la cual se deberá elegir la que consideramos es la real al problema planteado.

- Si adelgazo, hago dieta
- Si no adelgazo, hago dieta
- Si hago dieta, no adelgazo
- Si hago dieta adelgazo
- Hago dieta o no adelgazo

Solución:

✓ Problema: identificar la respuesta correcta al problema presentado por Sandra

✓ Definir las proposiciones atómicas:

p: Adelgazo

q: Hago dieta

✓ Determinar la relación lógica

$p \vee \neg q$

✓ Crear la tabla de verdad

p	(q)	($p \vee \neg q$)
V	V	V
V	F	V
F	V	F



F	F	V
---	---	---

Ejercicios:

1. Realizar un mapa conceptual con la definición y características de un algoritmo
2. Basado en el siguiente ejercicio realizar los algoritmos propuestos a continuación:

```
Algoritmo: Cargar un celular
INICIO.
1. Buscar el cargador.
2. Agarrar el cargador.
3. Agarrar el celular.
4. Poner el enchufe en el celular
5. Conectarlo a tomacorriente.
6. Esperar a que se ponga a cargar.
7. Desconectarlo del tomacorriente.
8. Ponerlo en un lugar seguro.
FIN
```

- ✓ Se propone diseñar un algoritmo para apagar fritar un huevo
- ✓ Se propone diseñar un algoritmo para despichar una llanta.
- ✓ Se propone diseñar un algoritmo para ir a cine.

3. Teniendo en cuenta las fases para el proceso de diseño de algoritmos analizar el siguiente problema:

Don Maluma necesita saber que es más favorable para él si pagar su parqueadero por día o pagar una mensualidad teniendo en cuenta que va 4 días a la semana a trabajar. El valor por día está en 4000 y la mensualidad le vale 85000.

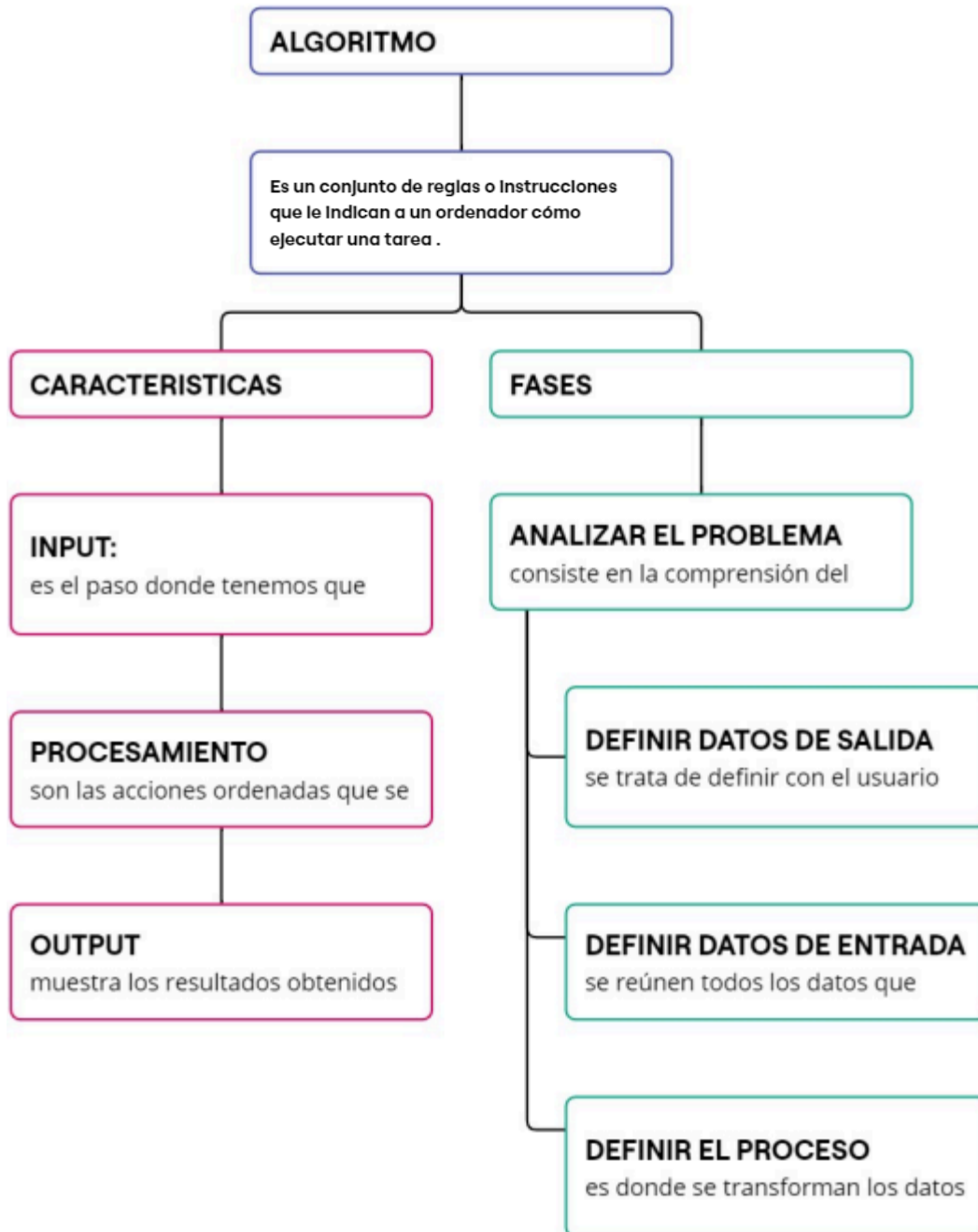
4. Ejercicio de lógica proposicional.

Analizar el siguiente enunciado “Se necesitan desarrolladores móviles que programen en el lenguaje Kotlin”



SOLUCION:

1.





2. Fritar huevo:

```
algoritmo para pritar un huevo:  
1 busco un huevo  
2 busco una paila  
3 busco el aceite  
4 pongo aceite en la paila  
5 pongo la paila en la estufa  
6 prendo la estufa  
7 pongo el huevo  
8 le echo sal  
9 apago estufa |
```

Despinchar una llanta :

```
algoritmo para despinchar una llanta:  
1 alistar herramientas  
2 meter el gato debajo del carro  
3 levantar el carro  
4 desatornillar la llanta  
5 bajar la llanta  
6 de arregla el neumatico  
7 se vuelve a montar la llanta  
8 atornillar la llanta  
9 bajar carro del gato  
10 fin|
```

ir a cine:

```
algoritmo para ir a cine  
1 escoger que pelicula quiere ver  
2 dirigirse al centro comercial  
3 comprar entradas  
4 comprar comida  
5 entrar a sala de cine|
```

3. Entender: Don maluma quiere calcular que le sale menos costoso, si pagar una mensualidad o seguir pagando a diario.

Definir salida: se calculará el valor más favorable para Don Maluma

Definir entradas:

- Don maluma trabaja 4 dias a la semana
- valor dia 4000
- valor mensualidad 85000

Proceso:

1 se calcula el valor por semana : 16.000

2 se el valor se multiplica por el número de semanas en un mes: 64.000



3 se compara el valor con el de la mensualidad

4 se llega a la conclusión que a Don Maluma le sale más barato pagar a diario.

4. - Identificar el problema: se necesitan desarrolladores móviles que programan en lenguaje kotlin

- descomponer:

p: Desarrolladores móviles

r: que programen en lenguaje kotlin

- Determinar:

p  r