**FUNDAMENTOS DE PROGRAMACIÓN**

**Definición**

Un algoritmo es un conjunto de procesos que permiten dar solución a un problema o necesidad

**Características del algoritmo**

* Definido
* Preciso
* Legible
* Finito

**Partes del algoritmo**

1. **Entrada:** Definir los datos de entrada del algoritmo
2. **Procesos:** Usan los datos de entrada
3. **Salida:** Resultado del procesos

**Desarrollo de algoritmo con pseudocódigo**

**¿Como resolver un problema con pseudocódigo?**

1. Tener un enunciado o problema claro, conciso y completo
2. Analizar los:
   1. **Datos de entrada**
   2. **Datos de salida**
   3. **Formulas que procesaran los datos**
   4. **Datos de prueba.**
3. Identificar las posibles soluciones

**VARIABLES Y TIPOS DE DATOS**

**Persistencia de datos**

El software son datos que se guardan en una memoria (ram o disco duro) en el caso de la ram son capacitadores llenos de energía, cuando se acaba la energía se acaban los datos.

* ¿Como se guarda la info. En la memoria RAM?

1. Se reserva el espacio, ejemplo un caracter
2. Se utiliza el espacio reservado, esto a través de impulsos electrónicos (0 y 1)
3. Se libera el espacio no utilizado

**Tipos de datos**

Estos varían según el lenguaje de programación. Existen algunos generales.

1. **Numérico:** Todos los positivos, negativos, enteros. Se representan normalmente por Integer (int).
2. **Decimal:** Tienen un punto, se representa por float.
3. **Texto:** Son los caracteres que representan un símbolo, se representa normalmente por comillas simples o dobles, un numero puede ser un texto si esta entre comillas, se representa por String.
4. **Cadena de caracteres:** Es la unión de varios símbolos, por ejemplo “Hola numero 4!”
5. **Booleanos(bool):**  Son False o True, estos son si o no.

**Declaración**

Hay cuatro tipos de escritura para las variables:

1. **Camel case:** Primera en minuscula, las otras mayuscular. Ejemplo, dayOfTheMonth
2. **Pascal case:** Mayuscula intercalada entre palabras. Ejemplo, DayOfTheMonth
3. **Snake case:** Palabras separadas por guion bajo. Ejemplo, day\_of\_the\_month
4. **Kebab case:** Palabras son separadas por guion. Ejemplo, day-of-the-month

Existen también **constantes**, estos siempre se **escriben** con **mayuscula sostenida**

**OPERADORES**

**Operadores aritméticos**

Son:

* Suma +
* Resta -
* Multiplicación \*
* División /
* Modulo (residuo) %

**Jerarquía de los operadores**

A continuacion se muestra como es la jerarquia para solucionar los problemas presentados:

1. **Signos de agrupacion {} [] ():** La solucion correcta para resolver es de adentro hacia afuera. Ejemplo: 9−{6+[5−3+(6−2)]+8(4−7)} Aqui se soluciona primero el parentesis, despues los corchetes y por ultimo las llaves.
2. **Potencias y raices:** Una vez se solucionen las agrupaciones y las operaciones con signos de agrupacion (ejemplo: 3(5)=15) se solucionan las raíces y potencias
3. **Multiplicaciones y divisiones:** Se solucionan de izquierda a derecha las multiplicaciones y luego las divisiones
4. **Sumas y restas:** Por ultimo se solucionan de izquierda a derecha primero las sumas y luego las restas

**Agrupacion de operadores**

Se hace con (). Ejemplo (4+2)\*(5-2)=18

Existen comparaciones dentro de las operaciones, da como respuesta booleanos.

* Mayor >
* Menor <
* Igual asignacion =
* Mayor igual >=
* Menor igual <=
* Igualdad comparación ==
* Diferente =!

Tambien existe agrupacion por OR y AND y tambien dan resultado booleano. Ejemplo, si queremos conocer la población mayor de 18 y menor de 65 usamos la siguiente linea:

edad >= 18 && edad <= 65

Si quisiéramos saber quienes son menores de 18 o mayores de 65 usamos OR

edad <= 18 || edad >= 65

**CONDICIONALES**

**Condicionales simples**

Son estrucutras