

	GIMNASIO SABIO CALDAS (IED) Nuestra escuela: una opción para la vida GUÍAS DE APRENDIZAJE – PLAN ESCOLAR	Código	PENP - 01
		Versión	001
		Fecha	18/03/2020
		Proceso	Gestión Académica

DOCENTE	Yudi Adriana Soler Franco	Grado	NOVENO
ASIGNATURA	PROGRAMACIÓN		
Correo electrónico de contacto	yudi.soler@sabiocaldas.edu.co		
Periodo académico	Tercer Periodo		
Tiempo de ejecución de la actividad	23 de agosto al 3 de septiembre		
¿Qué competencia(s) debo alcanzar?	Solución de problemas con tecnología Identifico y comparo ventajas y desventajas en la utilización de artefactos y procesos tecnológicos en la solución de problemas de la vida cotidiana.		
Temáticas mediadoras	APLIQUEMOS LO APRENDIDO - Solución de ejercicios prácticos - Ciclo for y while		
Metas	Socio-afectiva: Soy responsable y autónomo en la realización de mis actividades académicas, adaptando buenos hábitos de estudio.		
	Metas de aprendizaje: Soluciona ejercicios prácticos de los ciclos while y for, a partir de lo visto en clase.		

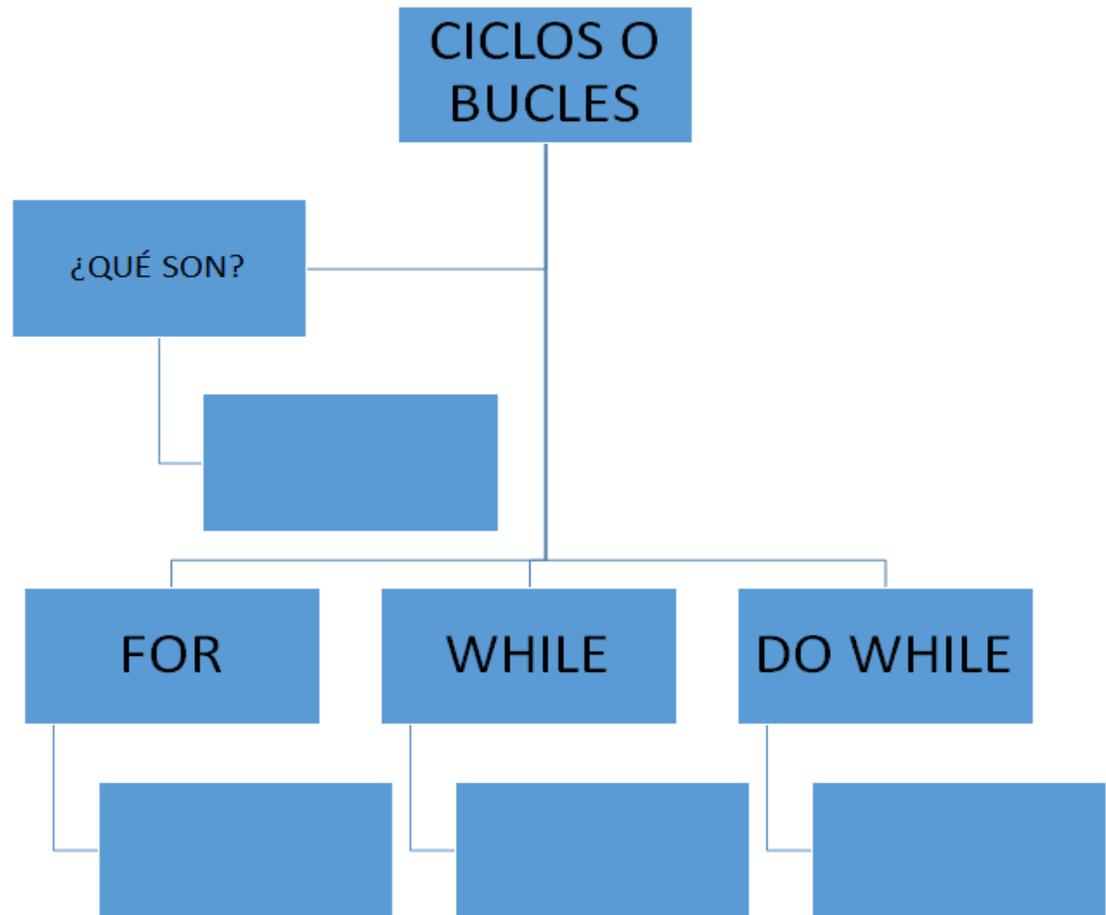
CRITERIOS DE EVALUACIÓN:

¿QUÉ SE VA A EVALUAR?	¿CÓMO SE VA A EVALUAR?	¿CUÁNDO SE VA A EVALUAR? Fechas
Comprender y aplicar los conceptos vistos en clase, para la solución de algoritmos a través del programa Pseint.	Resolución de problemas dados, a través de los procesos vistos en clase.	Primera semana:
		Segunda semana

SEMANA 1 (9 al 13 de agosto)

ACTIVIDAD INICIAL:

Completa el siguiente mapa conceptual.



CONTEXTUALIZACIÓN:

Estructuras repetitivas en Programación

Las estructuras repetitivas en programación, también denominadas bucles, te van a servir para poder repetir ciertas sentencias de código, bien un número determinado de veces, bien mientras se cumpla una condición.

Los ordenadores son buenos haciendo tareas repetitivas: siguen exactamente las instrucciones dadas y lo hacen muy rápidamente. Pero imagina que tuvieras que escribir las mismas instrucciones para cada repetición. Cuánto tiempo perderías, ¿no?. Además cada nueva repetición aumentarás las posibilidades de cometer un fallo al escribir. Y lo que es peor, imagina que luego quieres cambiar algo del

código que se repite, tendrías que hacerlo cada vez que lo has escrito.

Imagina que tienes que controlar un robot y tiene que andar en línea recta una serie de pasos. Podrías crear un código como el siguiente:

Andar paso

Andar paso

Andar paso

Andar paso

Eso haría que el robot andara 4 pasos. ¿Y si tiene que andar 10 pasos, repites 10 veces la misma línea? No es necesario, porque para eso tenemos las estructuras repetitivas (bucles). Así, podríamos escribir:

Repite 10 veces

Andar paso

Eso es genial, porque me evita tener que escribir un montón de código. Pero, ¿y si tiene que andar hasta llegar a una pared? No sabemos los pasos que tiene que andar. En ese caso existen estructuras repetitivas que permiten repetir tu código mientras una condición se cumpla.

Mientras no llegue a la pared

Andar paso

También podríamos expresarlo de otra forma, en la que nos aseguramos que al menos se anda un paso.

Repetir

Andar paso

Mientras no llegue a la pared

La diferencia entre el bucle 'mientras' y el bucle 'repetir mientras' es básicamente esa: en el 'mientras', si el robot está pegado a la pared no andaría, porque al

comprobar si ha llegado a la pared vería que sí y se pararía. Sin embargo en el bucle `repetir mientras` siempre andaría un paso, ya que comprueba si ha llegado a la pared al final.

La vida diaria está llena de ejemplos en los que podríamos aplicar estructuras repetitivas:

- Repite `decir Hola` hasta que alguien conteste (repetir-mientras)
- Repetir por siempre `inspirar y expirar` (condición que nunca es falsa)
- Mientras haya ejercicios de programación `hacer siguiente ejercicio` (mientras)
- Haz 10 repeticiones de abdominales (repetir con contador)

ACTIVIDAD DE AFIANZAMIENTO:

1. Teniendo en cuenta la lectura anterior, responde:

a. ¿Cómo crees que se puede implementar en la vida cotidiana los ciclos ?

2. Soluciona los siguientes ejercicios

A. diseñar un programa que escriba los números hasta el 5, haciendo uso de un ciclo.

B. Hacer un algoritmo para calcular la suma de los primeros cien números con un ciclo mientras.

VERIFICACIÓN DE APRENDIZAJES:

¿Qué comprendiste del tema visto?

¿Qué dudas tengo acerca del tema visto?

SEMANA 2 (16 de agosto al 20 de agosto)

ACTIVIDAD INICIAL:

CONTEXTUALIZACIÓN:

OTRA FORMA DE APLICAR LOS BUCLES, DESDE EL PROGRAMA SCRATCH

Bucles en Scratch

Podemos realizar bucles con Scratch igual que con BASIC. Estos bloques pertenecen a los bloques de **Control**

BUCLES

SCRATCH

BASIC

por siempre :

Este bloque es muy utilizado al igual que su equivalente en BASIC (que nosotros usamos para el BUCLE PRINCIPAL) muchos bloques no funcionan bien si no se actualizan constantemente. Por eso si los incluimos en un bucle infinito como este comienzan a funcionar correctamente. →



repetir hasta que < sensor / operación lógica > :

Repetir mientras o hasta que se cumpla una condición de un sensor o de una operación lógica. →



repetir (un número):

Repetir un número de veces, que podemos introducir con el teclado, puede ser una variable de **datos** ser cualquier valor resultado de **operaciones** o cualquier valor físico (sonido, tiempo, distancia,etc).



Fijate que los bloques que son **bucles** siempre llevan esta flechita

```
Suma_Numeros_Mientras
Definir suma,conta Como Entero;
conta <- 0; suma <- 0;
Mientras (conta<100) Hacer
  cont
```

ACTIVIDAD DE AFIANZAMIENTO:

Ingresa a la página <https://scratch.mit.edu/> y realiza el siguiente ejercicio, haciendo uso de los visto en clase sobre bucles. Con las indicaciones dadas en clase.

- Programa 1: Pinta 4 Cuadrados
- Programa 2: Pinta 8 Cuadrados
- Programa 3: Pinta 12 Cuadrados
- Programa 4: Pinta 24 Cuadrados
- Programa 5: Pinta 72 Cuadrados

VERIFICACIÓN DE APRENDIZAJES:

¿Qué tanto aprendí del tema?

¿Qué dudas me surgen del tema?

REFERENCIAS: WEBGRAFÍA.

<https://www.importancia.org/algoritmos.php>

<https://www.aprenderaprogramar.pro/2017/07/ejercicios-algoritmos.html>

[http://agrega.juntadeandalucia.es/taller/abranun984/03122015/ODE-e64ff0b7-a6b5-372d-8a95-fc22bb3ca08e/bucles en scratch.html](http://agrega.juntadeandalucia.es/taller/abranun984/03122015/ODE-e64ff0b7-a6b5-372d-8a95-fc22bb3ca08e/bucles_en_scratch.html)