

NAME Harald Nieto	CLASS Programación	SPEAKER Carlos Richard	DATE & TIME 29/8/22
----------------------	-----------------------	------------------------------	------------------------

Title *Algoritmos, diagramas y programas*

**Keyword**

- Operando
- Comparar
- Variables

**Topic**

*operadores relaciona-  
les, lógicos y coma*  
Los operadores relacionales  
se usan para comparar  
dos operandos de cualquier  
clase, ya sea números, caracte-  
res, variables, etc.

**Questions**

- ¿cumple  
- cuando cum-  
- plen los op-  
- eradores?

Por otra parte, los lógicos permiti-  
ten formulas, condiciones, com-  
plejas a raíz de condiciones  
simple, son de conjunción,  
disyunción y negación.

La coma, utilizada como opera-  
dor, sirve para encadenar di-  
ferentes expresiones de las  
variables dadas.

Cada uno de los operadores  
tiene prioridad sobre los otros  
a pesar de la evaluación de C.

**Summary:**

Los operadores cumplen la función de  
comparar los operandos y formulas con-  
diciones complejas.

NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Programación	Caplo Richard	29/4/22

Title *Algoritmo, diagrama y programas*

Keyword	Topic
<ul style="list-style-type: none"> <li>• serie</li> <li>• Procedu- rales</li> <li>• Pasos</li> <li>• Condicio- nes</li> </ul>	<p><i>Problema y algoritmo</i></p> <p>Los algoritmos son una serie de pasos, al igual que los pasos que seguimos los humanos, los cuales son procedurales, muchas veces se aplican los algoritmos de manera inventada cuando se ha enfrentado con el problema varias veces.</p>
<p>Questions</p> <p>¿qué relación tiene con la realidad?</p> <p>¿cómo es para los humanos?</p>	<p>Los humanos aplicamos todo eso todos los días, por ejemplo a la hora de abrir una puerta, solo que muy difícilmente para ser humano, pero para una máquina es fácil, pero una máquina requiere tener los pasos identificados y de pasos que hacen como si cumplen las condiciones.</p>

Summary: *Vemos como los algoritmos, más allá del código, pueden ser relacionados con aspectos de nuestra realidad.*

NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Programación	Carlos Richard	29/9/22

Title Algoritmos, diagramas y programas

Keyword	Topic
<ul style="list-style-type: none"> <li>Esquema</li> <li>Figuras</li> <li>Conexión</li> </ul>	<p>Diagramas de flujo</p> <p>Estos son la esquematización de lo que se sigue un algoritmo para dar la solución a un problema.</p> <p>La construcción del mismo es muy importante, ya que a raíz de esto se escribe el programa en lenguaje de programación, esto quedará ser representado con distintas figuras.</p>
<p>Questions</p> <p>¿qué elementos los son? ¿cómo se forman? ¿por qué es que importa tanto en el código?</p>	<p>El óvalo se usa para marcar el inicio y el fin, el rectángulo se usa para cambios en el flujo, el rombo se usa para operaciones condicionales de "SI O NO", el círculo se usa para conectar en la misma página, así como más figuras menos usadas.</p>

Summary: Los diagramas de flujo son la parte más importante de la programación, ya que trazan la ruta de la computadora.



NAME Harald Natio	CLASS Prague mación	SPEAKER Carlos Richard	DATE & TIME 29/4/22
----------------------	---------------------------	------------------------------	------------------------

Title Algoritmos, diagramas y programas

Keyword	Topic
<ul style="list-style-type: none"> <li>• Grupo</li> <li>• Memoria</li> <li>• Acceder</li> </ul>	<p>Tipo de datos</p> <p>Este dato se clasifica en simple y estructurado, lo simple solo pueden acceder a una sola casilla de memoria, mientras que lo estructurado pueden acceder a un grupo de casilla a la vez.</p> <p>Int: para los enteros</p> <p>Float: para los reales.</p> <p>Long: entero de largo alcance.</p> <p>Double: Real de doble precisión.</p> <p>Char: caracteres, no números.</p>
<p>Questions</p> <p>¿cuantos tipos hay?</p> <p>¿que es lo que los diferencia?</p>	

Summary: En C existen varios tipos de datos que se distinguen por el tipo de variable que manejan.

NAME	CLASS	SPEAKER	DATE & TIME
Harold Matto	Programación	Carlos Richard	29/9/22

Title Algoritmo, diagrama y programas

### Keyword

- Almacénados.
- Asignados

### Topic I identificadores

Los datos que vamos a usar en la computadora, sean simples o estructurados, deben ser almacenados en una celda de memoria para su uso posterior.

Para reconocer las celdas asignadas se les coloca un nombre o identificador, el cual al ser una letra, digamos A, y un carácter subrayado 1, digamos A1, siempre se le da un espacio por una letra, y generalmente los nombres no van de los 7 caracteres, aunque esto depende del compilador.

### Questions

¿Cómo distinguimos entre un dato y una letra?

### Summary:

Los identificadores, en síntesis, son nombres asignados a datos almacenados, para su fácil acceso.

NAME  
Harold Matto

CLASS  
Pragmatism

SPEAKER  
Capio  
Richard

DATE & TIME  
29/4/22

Title

Algoritmos, diagramas y programas

Keyword

• Cambio  
• Identifi-  
cadores  
• Ejecución

Topic

Constante y Variable

Questions

¿En qué  
se distin-  
guen?  
¿Cómo se  
nombran?

Por un lado las constantes  
son datos que no cambian  
durante la ejecución del pro-  
grama, para nombrarlas  
usan identificadores y existen  
tanto tipos de constantes  
como tipos de datos hay.

Mientras que por otra parte  
tenemos a las variables, que  
por definición son los datos  
que pueden cambiar duran-  
te la ejecución, para nom-  
brarlas igual usan iden-  
tificadores y por lo general  
se declaran en el programa  
principal y en las funciones.

Summary:

Las constantes y las variables  
son de las partes del código más im-  
portantes, dado su enorme rango de  
uso, incluso en la realidad.



NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Pragmatism	Carlos Richard	29/9/22

Title: Algoritmos, diagramas y programas

Keyword	Topic
<ul style="list-style-type: none"> <li>Numero</li> <li>Tabla</li> <li>operandos</li> </ul>	<p>Operadores aritmeticos y simplificados.</p> <p>No permite realizar operaciones en tu numero, constantes o variables, el resultado de una operacion aritmetica siempre es un numero y dado que el lenguaje C, distingue los tipos de operandos, se debe de tener en cuenta una tabla.</p>
<p>Questions</p> <p>¿que pasa si se desmemoria cada uno?</p>	<p>Tenemos que el lenguaje C, tiene una forma de simplificar los operandos que es bastante sencilla, de lo cual surge otra tabla o tener en cuenta para los operadores simplificados.</p>

Summary: Los operadores basicamente son un proceso que permite transformar cierto dato.

NAME Harold Mato	CLASS Programación	SPEAKER Carlos Richard	DATE & TIME 29/4/22
---------------------	-----------------------	------------------------------	------------------------

Title: Algoritmos, diagramas y programas

Keyword	Topic
<ul style="list-style-type: none"> <li>• Simplicidad</li> <li>• Valores</li> </ul>	<p>Operaciones de incremento, decremento y lógicas.</p> <p>Las operaciones de incremento y decremento son indispensables en el lenguaje, ya que simplifican la escritura de los programas. Se pueden usar antes o después de la variable, pero los resultados son diferentes.</p>
<p>Questions</p> <p>¿qué es lo que facilita cada una?</p>	<p>Las expresiones lógicas o booleanas están formadas por números, constantes o variables y operadores lógicos o relacionales, pueden poseer valores falso o verdadero y suelen usarse en estructuras repetitivas.</p>

Summary: Son simplificaciones muy útiles para el lenguaje.



NAME Harold Mateo	CLASS Prácticas	SPEAKER Carlos Richard	DATE & TIME 29/9/22
----------------------	--------------------	------------------------------	------------------------

Title Algoritmos, diagramas y programas

Keyword	Topic
<ul style="list-style-type: none"> <li>• Instrucciones</li> <li>• Resultado</li> <li>• Reglas</li> <li>• Solución</li> </ul>	<p><u>Programas</u></p> <p>Un programa es un conjunto de instrucciones que sigue la computadora para alcanzar un resultado, y esto debe ser escrito en un lenguaje de programación.</p> <p>A partir de un diagrama de flujo, el lenguaje está compuesto por un conjunto de reglas sintácticas que especifican la formación de las instrucciones válidas.</p> <p>Cada lenguaje de programación está estructurado, por tanto incorpora soluciones apropiadas de forma estructurada.</p>
<p>Questions</p> <p>¿Cómo se forma un programa?</p>	

Summary: Los programas se componen de código y su principal función es la de solucionar problemas.

NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Programación	Carlton Richard	29/4/22

Title *Estructura algorítmica*

Keyword	Topic
<ul style="list-style-type: none"> <li>Flujo</li> <li>Camino</li> <li>Ramas</li> <li>Decisión</li> </ul>	<p><i>Selectiva simple, doble y múltiple</i></p> <p><i>La estructura selectiva permite que el flujo de un diagrama siga por un camino específico si cumple las condiciones de verdadero o falso. Se puede dar o no la explicación.</i></p> <p><i>La selectiva doble permite que el camino se bifurque en dos ramas al momento de tomar la decisión, teniendo dos operaciones distintas si es verdadero o falso.</i></p> <p><i>La múltiple al igual que la anterior bifurca el camino, solo que ahora pueden haber más caminos con otras operaciones.</i></p>
Questions	
<p><i>¿qué distingue a cada tipo de selectiva?</i></p>	

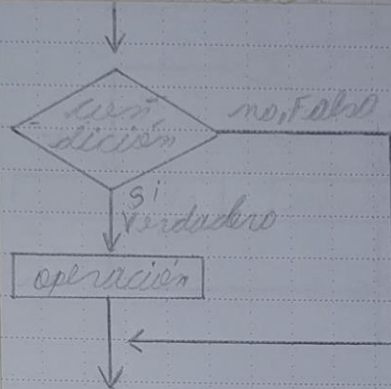
Summary: *Cada tipo de estructura permite resolver problemas por medio de distintos tipos de diagramas.*

NAME	CLASS	SPEAKER	DATE & TIME
Harold nato	Práctica mañana	Carlos Richard	29/4/22

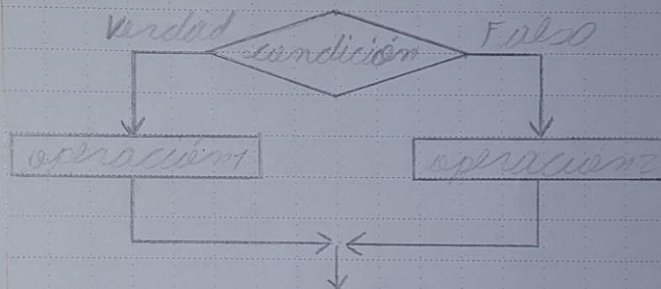
Title *Estructura algorítmica selectiva*

Keyword

Topic *Estructura selectiva  
Simple de si*



Questions



Summary:



NAME Harold Mateo	CLASS Programación	SPEAKER caso Richard	DATE & TIME 29/9/22
----------------------	-----------------------	----------------------------	------------------------

Title *Estructuras algorítmicas*

Keyword

Topic

*selectiva múltiple o switch*

```

graph TD
    Entry(( )) --> Selector{selector}
    Selector --> Accion1[acción 1]
    Selector --> Accion2[acción 2]
    Selector --> Accion3[acción 3]
    Accion1 --> Join(( ))
    Accion2 --> Join
    Accion3 --> Join
    Join --> Exit(( ))
  
```

Questions

Summary:

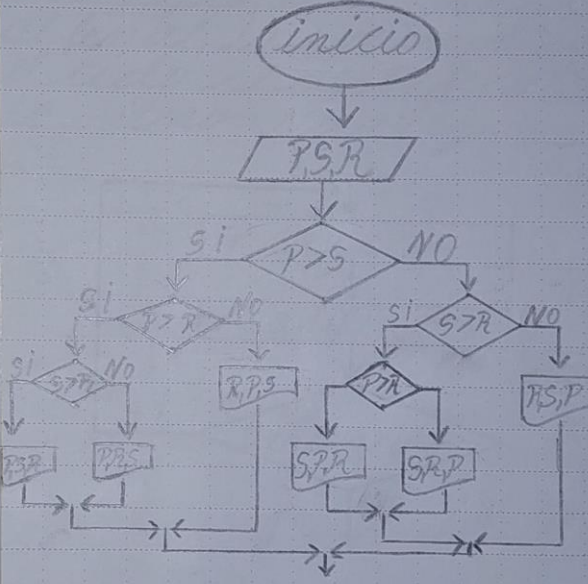
NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Pragmática	Carlo Richard	29/9/22

Title *Estructuras algorítmicas*

**Keyword** *Decision  
Repetir*

**Topic** *Estructuras selectivas en cascada*  
*En el proceso de solución de un problema, por lo general, se elige el camino, a veces es necesario tomar otra decisión y se repite varias veces.*

**Questions**  
*¿qué tan complejas pueden ser las estructuras en cascada?*



**Summary:** *Las estructuras en cascada permiten solucionar problemas de complejidad más alta.*

Title *Estructuras algorítmicas repetitivas*

<p><b>Keyword</b></p> <ul style="list-style-type: none"> <li>• Evalúa</li> <li>• Principio</li> </ul>	<p><b>Topic</b> <i>Estructura repetitiva do-while.</i></p> <p>Es otra estructura que se encuentra en casi todos los demás lenguajes, siendo similar al "replat" pero a diferencia de sus contrapartes en las que se evalúa al principio, estas se evalúan al final, lo que implica que dado se debe ejecutar por lo menos una vez.</p>
<p><b>Questions</b></p> <p>¿Dónde se ejecuta?</p> <p>¿Cuántas veces se ejecuta?</p>	<pre> graph TD     Start(( )) --&gt; Procesos[Procesos]     Procesos --&gt; Modificación[Modificación]     Modificación --&gt; Evaluación{Evaluación}     Evaluación -- SI --&gt; Start     Evaluación -- NO --&gt; Exit(( ))   </pre>

**Summary:**

.....

.....

.....

.....



NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Programación	Carlos Richman	29/9/22

Title *Estructura algorítmica repetitiva*

Keyword

*Conjunto*

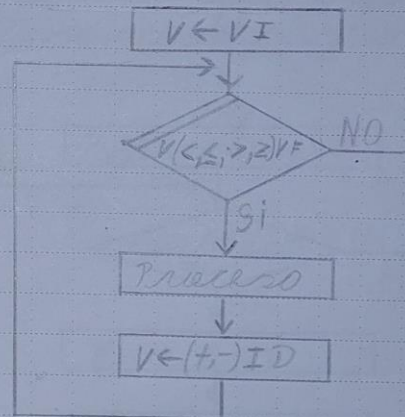
Topic

*Estructura repetitiva FOR*

*Es una estructura lógica  
algorítmica usada para rep-  
etir un conjunto de instrucc-  
iones varias veces y se encuen-  
tra en casi en toda los lenguajes  
de programación.*

Questions

*¿Es más tan-  
to los lenguajes  
de programación?*



Summary:

NAME  
Harold Matéo

CLASS  
Programación

SPEAKER  
Carlo  
Richard

DATE & TIME  
29/9/22

Title Estructura algorítmica repetitiva

### Keyword

• Cantidad  
• Proposición  
• Cíelo  
• Comprobar

### Topic

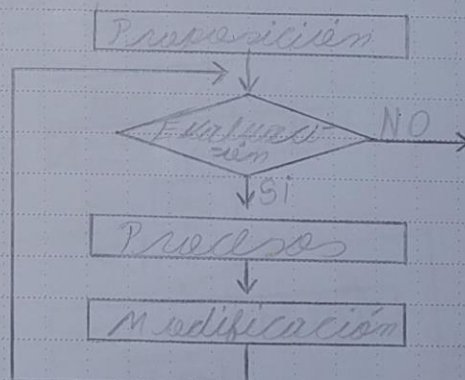
Estructura repetitiva

While

Es muy similar a "for" con la diferencia de que en esta estructura, el número de veces que se debe repetir, depende de la cantidad de proposiciones que haya en el ciclo, y en cada ciclo se comprueba que la condición sea diferente de cero, si no lo es, se detiene.

### Questions

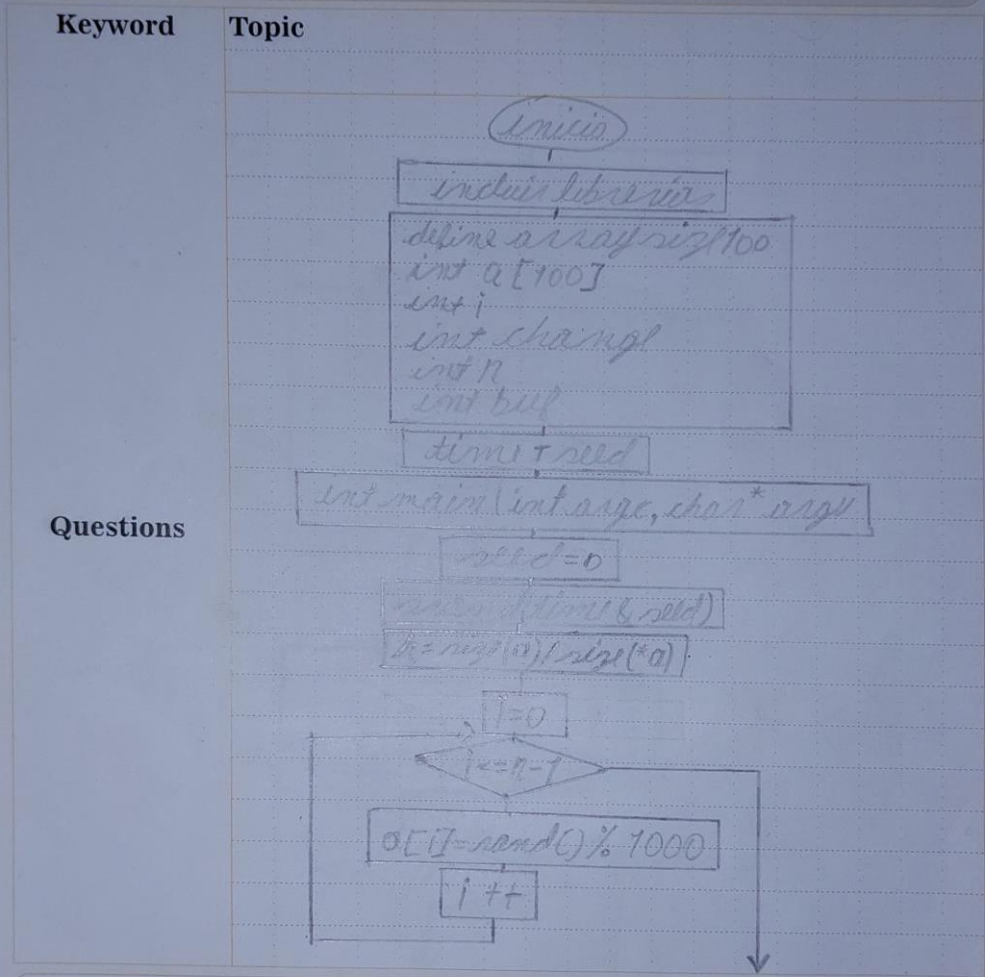
¿Por qué se  
llaman a estas  
estructuras  
repetitivas?



### Summary:

Veremos que es importante que haya diferentes estructuras de repetición, dando que hay problemas que requieren un algoritmo distinto.

Title: *Código de ejemplo*



Summary:



NAME	CLASS	SPEAKER	DATE & TIME
Harold mato	Programación	Richard	13/10/22

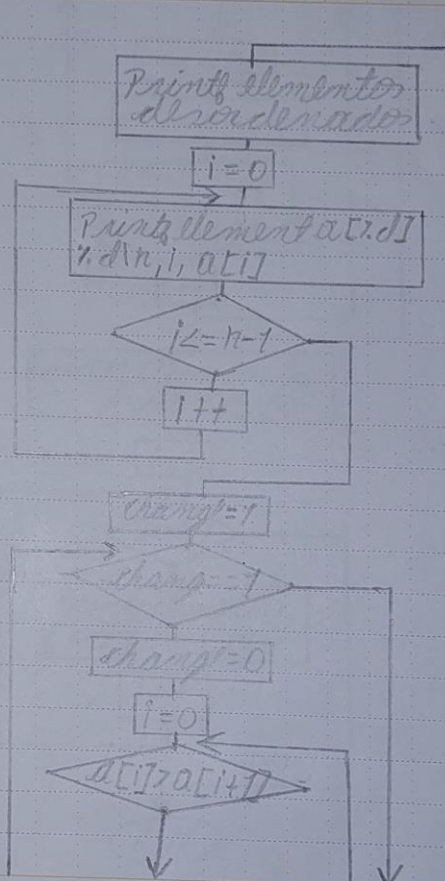
Title

Código de ejemplo

Keyword

Topic

Questions



Summary:

NAME	CLASS	SPEAKER	DATE & TIME
Harold Matos	Programación	Richard	13/10/22

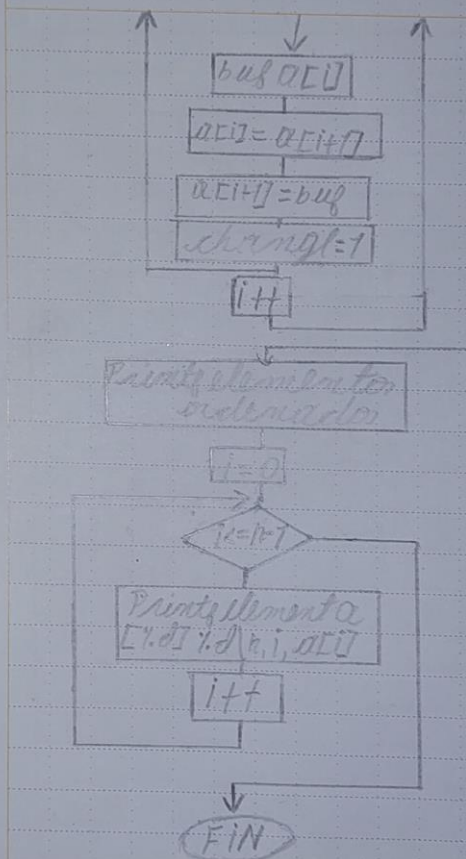
Title

Código de ejemplo

Keyword

Topic

Questions



Summary:

NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Programación	Carlos Pichardo	19/10/22

Title *PRO-Git*

Keyword	Topic
<ul style="list-style-type: none"> <li>• Archivos</li> <li>• Centralizado</li> <li>• Local</li> </ul>	<p><i>Fundamentos de git</i></p> <p>Esto puede ser un control de versiones de los archivos, pero también esto puede ser un controlizado o descontrolizado.</p> <p>Cada toda vez que se realiza una operación se realiza una copia local y puede ser un sistema de versiones de la integridad.</p>
<p>Questions</p> <ul style="list-style-type: none"> <li>• ¿Qué es?</li> <li>• ¿Qué es?</li> <li>• ¿Qué es?</li> <li>• ¿Qué es?</li> <li>• ¿Qué es?</li> </ul>	<p>La parte más importante son los estados de un documento, ya que puede perderse la información, y esto son los conceptos de trabajo, el stage y el repositorio siendo sus ubicaciones.</p>

Summary: *Git es un programa que ejecuta a nivel local y de consola, que permite tener un control de los programas*



NAME	CLASS	SPEAKER	DATE & TIME
Harold Matio	profesora materia	Richard	14/10/22

Title PRO-Git

Keyword .instalar  
.identificar

Topic Linia de comandos

Esta es solo una de las muchas formas que existen de usar git, siendo de sus funciones mas fundamentales. siendo necesario usarla si se desea instalar en linux, teniendo que usar el comando "apt-get install git".

Questions  
¿cómo se  
usa en  
linux?

siendo utilizada tambien para identificar la primera vez que se crea un archivo a github.

Summary: Ademas de todo, git se puede usar en casi cualquier sistema.

NAME	CLASS	SPEAKER	DATE & TIME
Harold Walter	Prague macion	Carlos Richard	19/10/22

Title

PRO-Git

Keyword

Topic

Comandos simples

Questions

"Git init" es uno de los comandos, siendo el primer paso para iniciar un repositorio.

"Git clone" nos permite crear un repositorio remoto a lo local como copia.

"Git add" permite dar seguimiento a los archivos.

"Git commit" permite guardar los cambios en el repositorio.

"Git status" nos ofrece un listado de los cambios de la rama en la que estamos trabajando.

Summary:

NAME	CLASS	SPEAKER	DATE & TIME
Harold mator	Prograx -miller	Gilberto Richard	19/10/22

Title *Pro-Git*

Keyword

Topic

*Comandos interme-  
dios.*

*"Git rm" sirve para la eli-  
minación de un archivo.*

*"Git log" permite mirar atrás  
en los puntos de guardado.*

*"Git commit -- amend" es  
la función de retroceder en  
tres puntos de guardado.*

Questions

*"Git checkout" permite el  
cambio de rama de trabajo.*

*"Git remote" permite ver los  
repositorios remotos creados.*

*"Git fetch" trae desde el repo-  
sitario remoto, los archivos  
que no están en local.*

Summary:



NAME	CLASS	SPEAKER	DATE & TIME
Harold Mateo	Pregrado -materia-	Carlos Richard	19/10/22

Title *Pro-Git*

Keyword

Topic *Comandos alternos*

"Git merge" permite unir  
-mas ramas de trabajo.

"Git branch" sirve para  
crear o eliminar ramas.

"Git tag" permite etiquetar  
los distintos commits.

Questions

Summary: