



Programación

Clase 3

División en subtareas

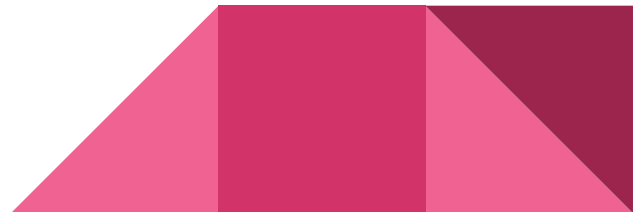
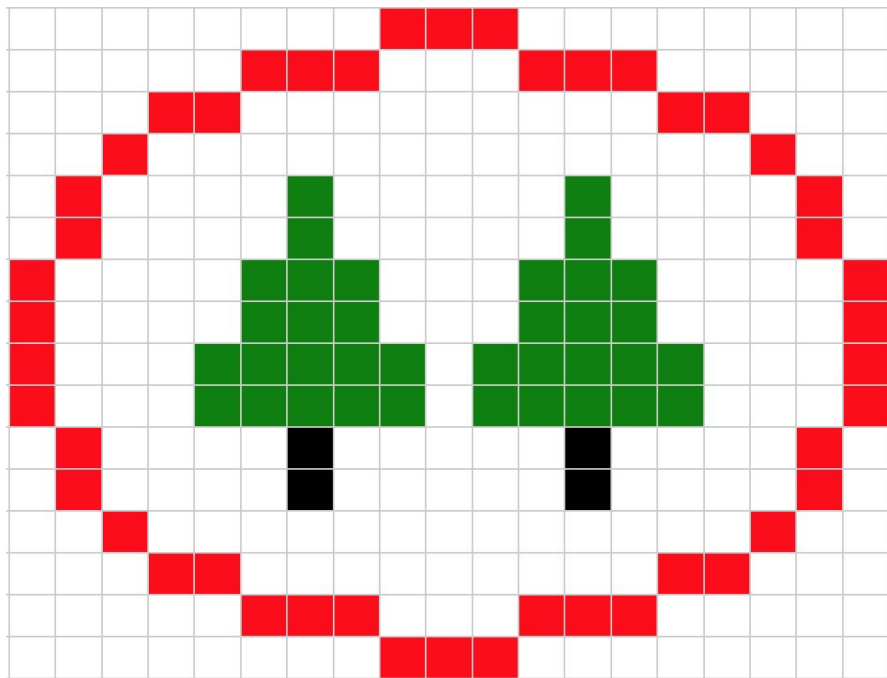
Universidad Nacional de Quilmes

Ejercitamos un poco



Ejercicio

Realicemos el siguiente dibujo en QDraw.



¿Por dónde arrancamos?



Recordemos siempre que **Programar es
comunicar**




Intentando comunicar la solución

Si le intentamos contar a alguien que es lo que hay que hacer, terminaremos diciendo:

arriba, arriba, arriba, pinta, arriba, pinta... etc.

Esta solución presenta varios problemas:

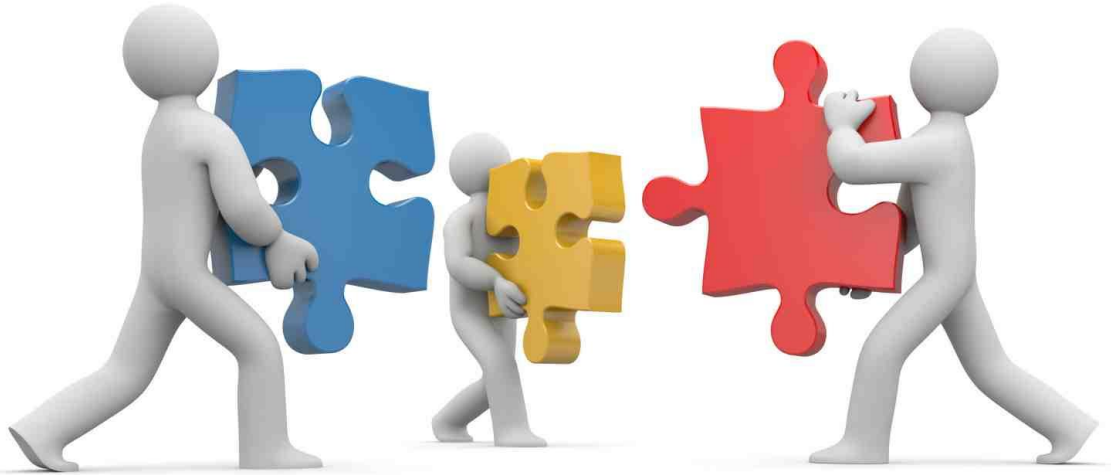
- El código es confuso hasta para nosotros mismos.
 - La otra persona no tiene idea de qué le estamos hablando.
 - Los comentarios pueden ayudar a entender el código, pero no solucionan el problema de fondo.
 - Uno quisiera poder transmitir la idea de una forma más sencilla.
- 

Lo que quisiéramos transmitir es algo al estilo:

```
programa {  
    dibujar óvalo externo en rojo  
    dibujar árbol derecho  
    dibujar árbol izquierdo  
    volver el cabezal a la posición inicial  
}
```



Divide y vencerás



Divide y vencerás

Al dividir el problema general en problemas más pequeños, podemos centrarnos en resolver cosas más sencillas, que requieren menos código y son más fáciles de razonar.

De esta forma es más fácil atacar problemas grandes, para arribar a una solución integral.



Divide y vencerás. Ejemplo 1

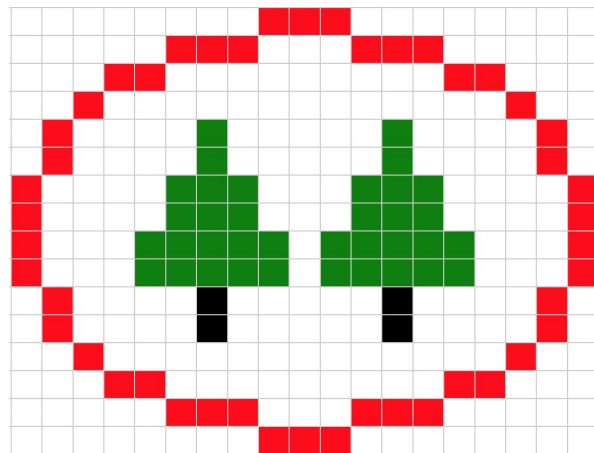
¿Qué pasos o acciones realizan cada vez que realizan la actividad de “cepillarse los dientes”?



Divide y vencerás. Ejemplo 2

Dibujar logo

- ↳ Dibujar círculo rojo
 - ↳ Arriba, arriba, derecha, pintar de rojo ...
- ↳ Dibujar árbol izquierdo
 - ↳ Dibujar copa del árbol
 - ↳ Pintar de verde, arriba, pintar
 - ↳ Dibujar tronco del árbol
 - ↳ Pintar de negro, arriba, pintar
- ↳ Dibujar árbol derecho
 - ↳ Dibujar copa del árbol
 - ↳ Pintar de verde, arriba, pintar
 - ↳ Dibujar tronco del árbol
 - ↳ Pintar de negro, arriba, pintar



Procedimientos




Procedimientos

Los procedimientos son una **forma de estructurar el código** para reflejar estos esquemas mentales que hemos comentado.

Un **procedimiento** es una **nueva instrucción definida por el usuario**.

Un procedimiento se define mediante la palabra “**procedimiento**”, seguida de un **nombre** (El cual no puede contener espacios y comienza con **mayúscula**), paréntesis vacíos y un **bloque de código**.



Procedimientos

```
procedimiento DibujarTroncoDeArbol () {
```

```
  /*
```

```
  PROPÓSITO: Dibuja el tronco de un árbol de color negro, de dos  
             celdas de alto, con la celda actual siendo la celda inferior del tronco.
```

```
  */
```

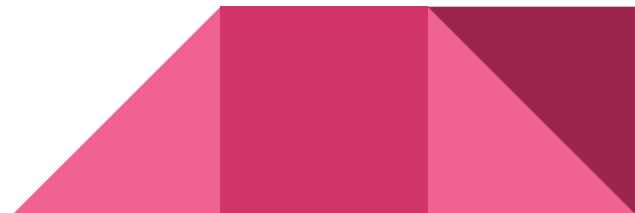
```
  PintarNegro
```

```
  MoverArriba
```

```
  PintarNegro
```

```
  MoverAbajo
```

```
}
```



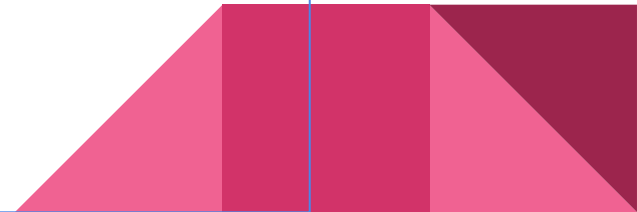
Ahora tenemos Instrucciones y Procedimientos

Acciones :Set limitado de instrucciones:

- MoverDerecha
- MoverArriba
- MoverIzquierda
- MoverAbajo
- PintarNegro
- PintarRojo
- PintarVerde
- Limpiar

Procedimientos

- DibujarTroncoDeArbol ()




Llamar a procedimientos

Los **procedimientos definidos** pueden ser **llamados** luego en cualquier lugar del programa.

Al momento de **ejecutar el programa**, en los lugares en los que se llama a un procedimiento se ejecuta el **bloque de código con el cual fue definido**.

Un procedimiento puede ser **llamado en cualquier bloque de código**, ya sea el del programa o el de otro procedimiento.

Un procedimiento **se define una sola vez**, pero se puede **llamar tantas veces como se desea**.



Llamar a Procedimientos

Para llamar a un procedimiento basta utilizar el nombre del mismo seguido de paréntesis dentro de un bloque de código.

```
programa {  
    DibujarTroncoDeArbol()  
    ...  
}  
  
procedimiento DibujarTroncoDeArbol () {  
    /* */  
    PintarNegro  
    MoverArriba  
    PintarNegro  
    MoverAbajo  
}
```



Llamar a Procedimientos

Pueden llamarse más de una vez, aunque se define una sola vez.

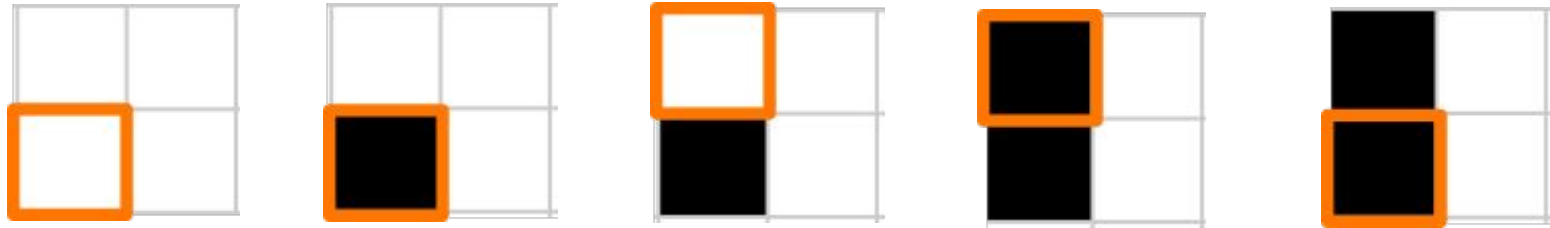
```
programa {  
    DibujarTroncoDeArbol()  
    MoverIzquierda  
    DibujarTroncoDeArbol()  
}
```



Ejecución del programa anterior

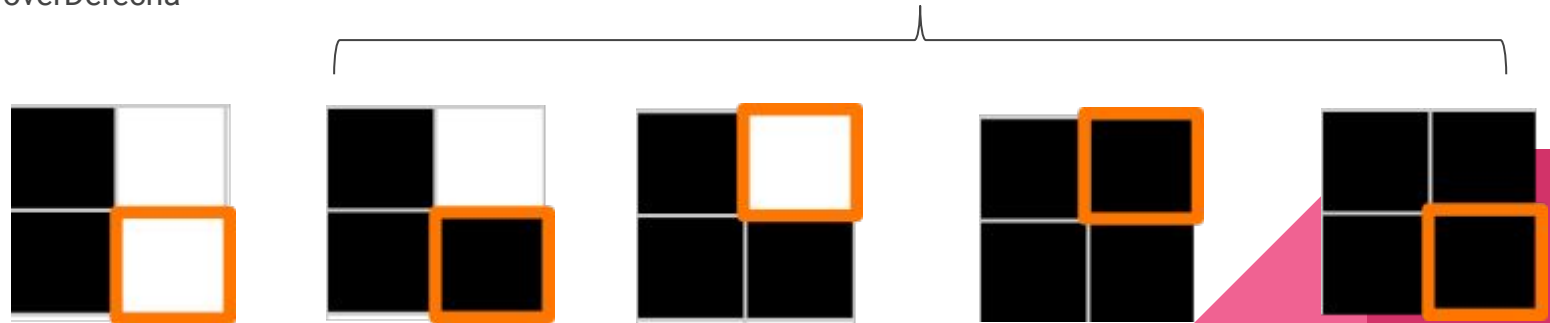
DibujarTroncoDeArbol()

Inicio



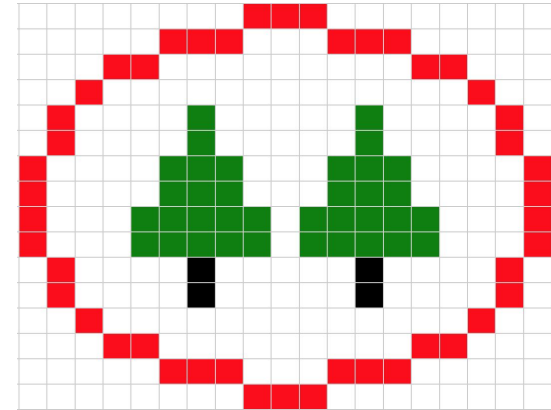
DibujarTroncoDeArbol()

MoverDerecha



Llamar a Procedimientos en otro Procedimiento

```
procedimiento DibujarArbol () {  
    DibujarTroncoDeArbol()  
    IrACopa()  
    DibujarCopaDelArbol()  
}
```



Cada procedimiento debería tener su propósito y comentarios

Dibujamos el árbol

```
programa{
```

```
  /* ... */
```

```
  DibujarArbol ()
```

```
}
```

Mi programa se resume a llamar al procedimiento principal, y solo eso.



Ejercitamos un poco con Lightbot



Nivel 1: Ejemplo

The image shows a game level editor interface. On the left is a 3D view of a level with a pink robot on a platform. The level is composed of grey and blue blocks. In the top left corner, there are icons for a play button and a speaker, and the text "2-1". At the bottom is a toolbar with icons for an arrow, a lightbulb, a left turn, a right turn, a stack of blocks, and a "P1" block.

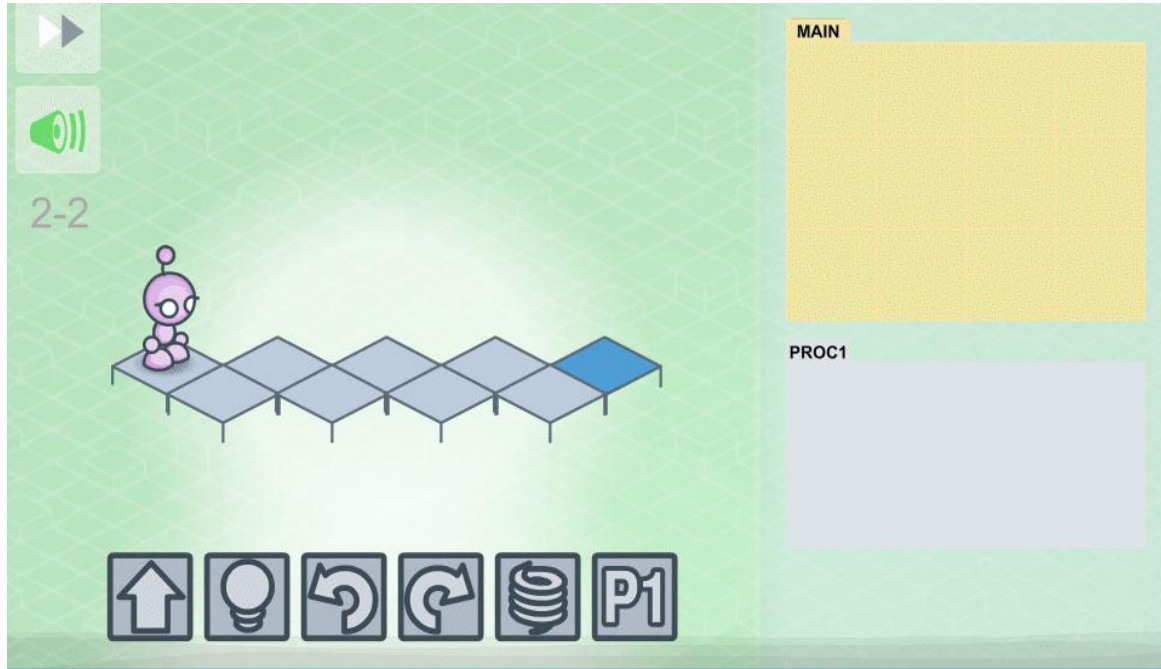
On the right is a code editor with two sections:

- MAIN**: A yellow background containing a sequence of blocks: "P1", a right turn, "P1", a right turn, and "P1". An orange arrow points from the "P1" block to the text "Llamada al procedimiento".
- PROC1**: A grey background containing a sequence of blocks: an arrow, an arrow, an arrow, and a lightbulb. An orange arrow points from the lightbulb block to the text "Descripción del procedimiento".

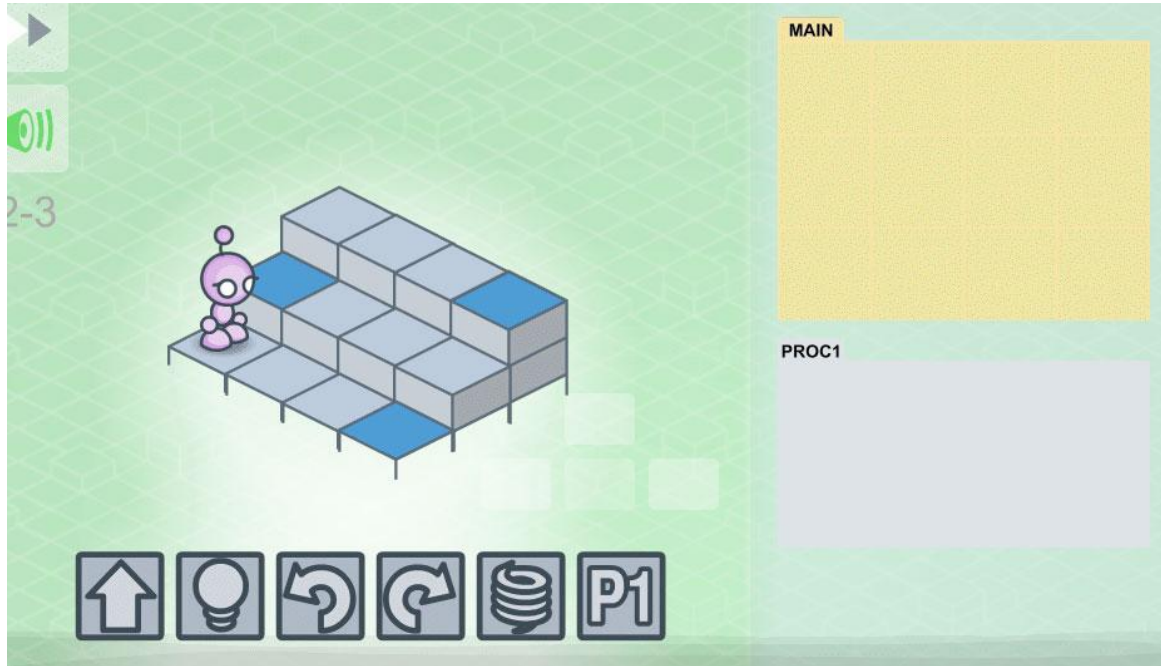
At the bottom of the code editor, there is a label "Cod P1" with a blue arrow pointing to the right.



Nivel 2



Nivel 3



Nivel 4

The image shows a 3D block-based programming environment. On the left, a pink robot stands on a structure of blue and grey blocks. The interface includes several control elements:

- Top left: A blue left arrow and a blue right arrow.
- Top center: A blue question mark icon and a green right arrow icon.
- Left side: A grey play button, a green speaker icon, and the text "2-4".
- Bottom: A row of icons including an upward arrow, a lightbulb, a left arrow, a right arrow, a stack of coins, and two boxes labeled "P1" and "P2".

On the right side, there is a yellow rectangular area labeled "MAIN" and two grey rectangular areas labeled "PROC1" and "PROC2", representing a workspace for writing code or logic.

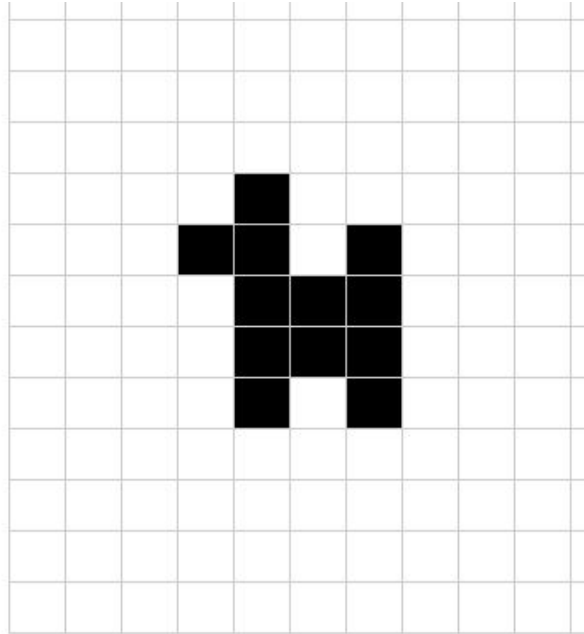


Ejercitamos un poco con QDraw



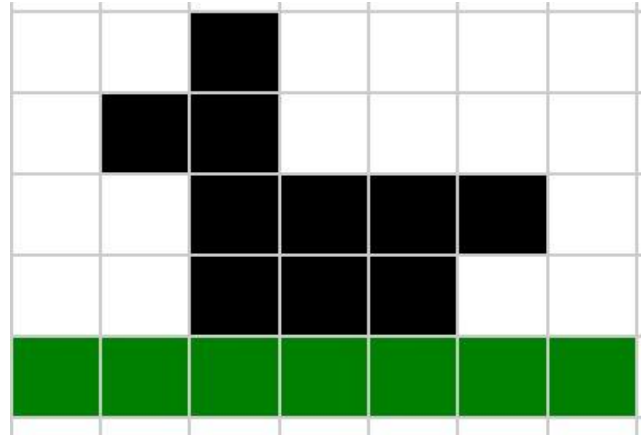
Actividad 1

Implemente un programa que realice el siguiente dibujo utilizando procedimientos.



Actividad 2

Implemente un programa que realice el siguiente dibujo utilizando procedimientos.



¿Encuentran algo parecido al dibujo anterior?



Programación

Clase 3

División en subtareas

Universidad Nacional de Quilmes