

La enseñanza de programación para educación primaria en establecimientos analógicos

Diego Fraiese
dfraiese@gmail.com

Resumen. Se examinará la experiencia realizada con alumnos de quinto grado a los cuales se le dictó un taller de programación de lenguaje Python sin que tengan ninguna experiencia previa en lenguajes y en un establecimiento donde no se disponía de computadoras. Se explicitarán los resultados de la aplicación de los enfoques de aprendizaje adaptativos y gamificación.

1. Introducción

La necesidad en la actualidad de introducir la enseñanza de programación como materia curricular se hace cada vez más evidente en la generación Z debido a su condición de nativos digitales, lo cual le permite manejar con cierta facilidad dispositivos electrónicos y la de afianzar, utilizando la abstracción que impone las técnicas algorítmicas, algunos conceptos curriculares los cuales en las evaluaciones se evidencian como poco asimilados por el estudiante, por tanto, la enseñanza de programación no resulta un fin en sí mismo, sino un medio para que se incorporen conceptos curriculares a través de la aplicación de estos en la programación. En este proceso se hizo imprescindible la incorporación de métodos de enseñanza específicos y adaptados a la ausencia de computadoras y por tratarse de una experiencia nueva en la escuela.

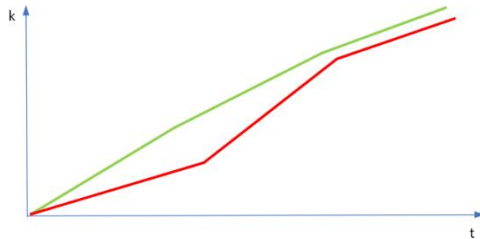
2. El proceso

Debido a la falta de computadoras se propuso la utilización de los celulares personal de los estudiantes a fin de utilizarlos como entornos de programación. Debido a la característica de nativos digitales de estos niños, la instalación de una app como *pyDroid* fue sumamente sencilla y no requirió esfuerzo alguno. Se decidió que el lenguaje de programación fuera Python debida a su sintaxis sencilla y su curva de aprendizaje rápida, también por su potencia ya que los niños por interés personal o institucional podían seguir avanzando con el lenguaje hasta niveles expertos si lo desearan. Esa era la diferencia fundamental con el lenguaje propuesto inicialmente, *Scratch*. Sus características basadas en bloques lo hacían sencillo de explicar, pero difícil de escalar hacia mayores demandas de los niños en años superiores.

Se dispuso la aplicación de dos métodos pedagógicos

a. Aprendizaje adaptativo

Mediante evaluaciones sencillas y reiteradas realizadas en el aula se fueron adaptando los ejercicios propuestos en dos diferentes agrupaciones, *sencillos*, para estudiantes con mayor dificultad en la incorporación de conceptos y *avanzados*, para aquellos más adelantados. Se propuso los primeros como obligatorios y los segundos como opcionales, estableciendo una curva de aprendizaje que alcanzara los contenidos mínimos esperados al final del taller mediante una convergencia de conocimientos.



En rojo puede verse la curva de aprendizaje de aquellos alumnos con más dificultades para incorporar los contenidos y en verde la de aquellos más adelantados. Mediante la división de tareas se logró la convergencia de conocimientos al final del taller sin que implicara ralentizar el aprendizaje de un grupo sino la de elevar el nivel de dificultad en forma más gradual a aquellos que tenían más dificultades.

b. Gamificación

Debido a la edad de los estudiantes se estableció un aprendizaje fundamentalmente basado en el juego y en la competencia por equipos, comenzando por los juegos de rol para explicar los rudimentos de la programación.

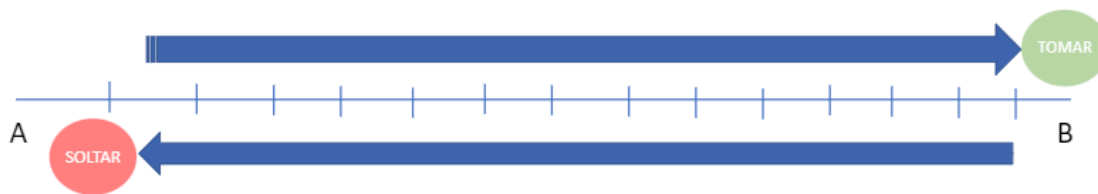
Las primeras clases se practicaron juegos de rol en donde los estudiantes debían guiar al profesor utilizando un conjunto finito de instrucciones o *set* que implicaba la incorporación del concepto de *instrucción* o *sentencia* en lenguajes de programación.

Por ejemplo, se establecieron las siguientes instrucciones

PASO
GIRO
TOMAR
SOLTAR
SALTAR

Los giros se establecieron en noventa grados en sentido de las agujas del reloj y tomar/soltar refería a la interacción con objetos. La referencia al salto no tenía ninguna otra connotación didáctica mas que fomentar lo lúdico del juego provocando el salto del profesor y las consecuentes risas tras la acción.

Se los dividió en grupos de cuatro estudiantes y se les pidió que diseñaran un conjunto de órdenes donde el profesor debía realizar cierto recorrido, tomar un objeto y depositarlo en otro lugar preestablecido.



El grupo que lograba diseñar el conjunto de órdenes mas óptimas, entendiéndose por ello el menor conjunto de órdenes para lograr el objetivo preestablecido se lo incorporaba en la lista de *honor* a modo de recompensa como el mejor grupo y así sucesivamente fomentando la competencia.

El primer juego realizado mostró algunas deficiencias en el planteo por parte del docente afianzando la horizontalidad de la educación. Los alumnos escribían las instrucciones como se les había enseñado en años de clases, de izquierda a derecha y no *top-down* como se requiere en un lenguaje de programación. El juego dio pie a la incorporación de un conocimiento fundamental, la forma de escribir un programa informático.

En las sucesivas clases se fue complejizando la dificultad colocando obstáculos y haciendo que el estudiante deba pensar una estrategia para rodear al obstáculo. Rápidamente el niño veía el resultado de su programa y podía corregirlo en una próxima iteración. El uso de la instrucción *saltar*, que provocaba la risa al ver a su profesor en ese ejercicio, dejó de ser tan frecuente para pasar a lograr la correcta optimización del código a fin de obtener la recompensa.

El próximo paso fue utilizar el mismo conjunto de instrucciones, pero en inglés de forma de dar el primer salto hacia el lenguaje Python.

STEP
JUMP
TURN
PUSH
POP

Por razones de carácter pedagógico se acordó con los estudiantes el uso de *push* y *pop* como traducciones de *tomar* y *soltar* ya que luego se utilizarían en la clase de *colas* y *pilas*.

El segundo gran paso fue la incorporación del celular para realizar pequeños programas funcionales fundamentalmente para explicar el uso de iteraciones, por ejemplo

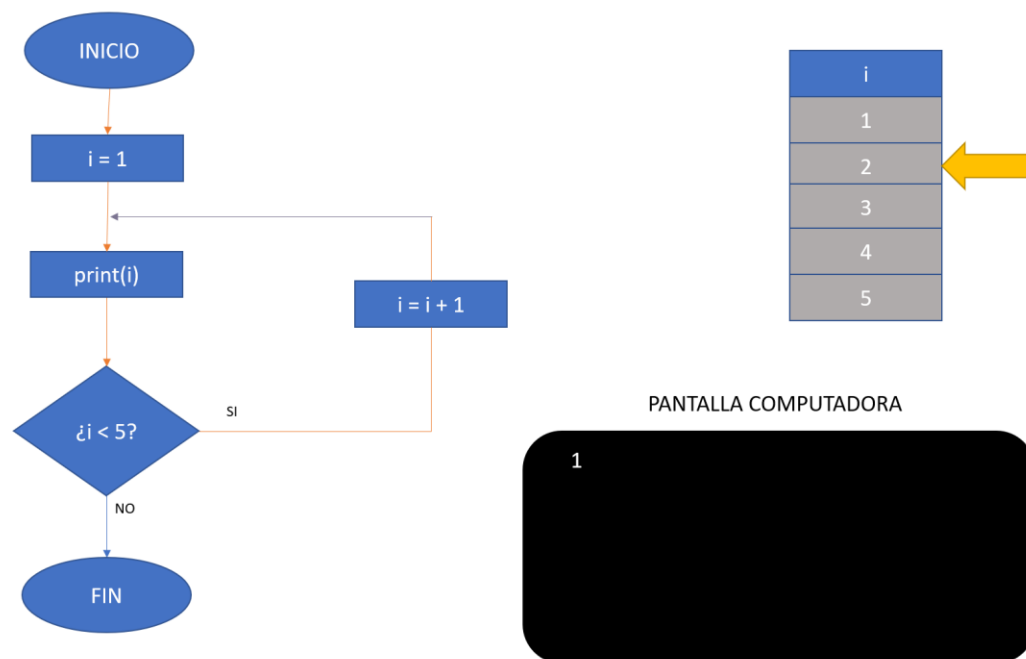
```
i = 1
while (i < 11)
    print('paso')
    i = i + 1
```

Allí surgió la necesidad de explicar conceptos de lógica y variables por tanto se utilizó el método pedagógico tradicional, los alumnos copiaban del pizarrón y resolvían los planteos lógicos en sus cuadernos

Se realizó una tabla muy sencilla que se fue complejizando a medida que el niño iba avanzando en la comprensión de las tareas propuestas

Condición	¿Verdadero?	¿Falso?
$5 < 11$		
$1 > 1$		
$3 > 2$		
$6 \leq 20$		
$20 \geq 1$		

Para explicar de una forma sencilla el accionar de los programas se diapositivas de Power Point de Microsoft como la siguiente, las cuales se proyectaban mediante el uso de una TV



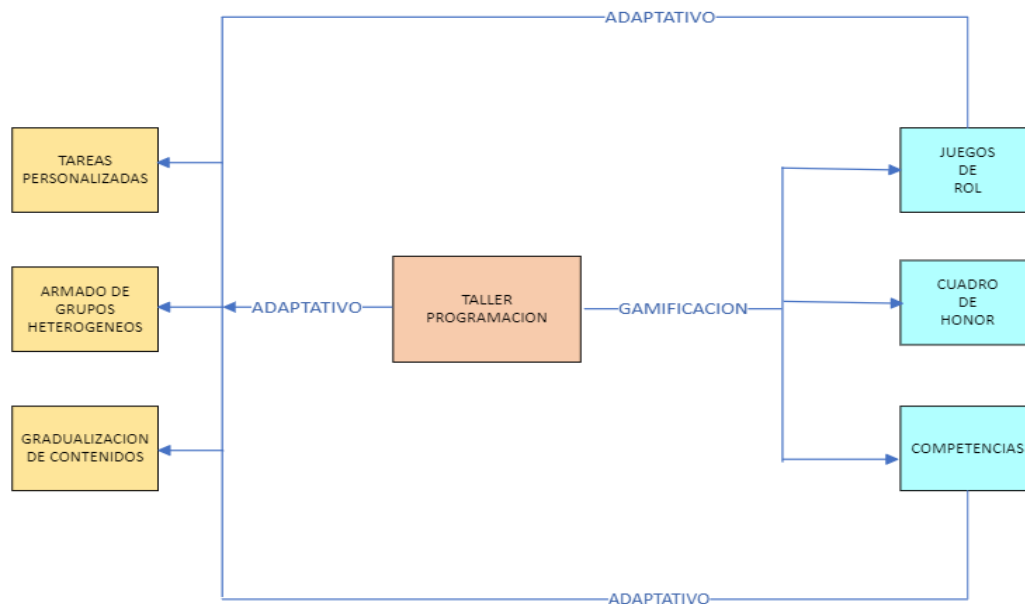
El próximo paso fue el de la utilización de los celulares. Cada estudiante copiaba el programa en la *Integrated Development Environment -IDE- PyDroid* y la ejecutaba, quienes tenían problemas en la ejecución se los agrupaba con quienes habían logrado el objetivo de forma que puedan compartir experiencias. Al final de la clase se proponía un programa con dificultad mayor para su realización en el hogar.

Una de las competencias realizadas y con mas aceptación por parte de los estudiantes fue la de *Mi celular es el más rápido*, se proponía realizar un ciclo de cien mil ocurrencias, el celular que terminaba primero en mostrar la ocurrencia cien mil obtenía era designado *celular ultra poderoso*, rápidamente los niños comenzaron a explorar, sin saberlo, test de estrés a sus dispositivos colocándoles valores de millones de iteraciones.

Otro de los juegos fue Los agentes secretos, en donde se les enseñó la encriptación por el método de desplazamiento y mediante programas ejecutados en sus celulares debían *encriptar* y *desencriptar* palabras determinadas. El grupo que lograba hacerlo primero era incorporado en el *cuadro de honor*.

Esto propició otra forma de nivelación ubicando a los estudiantes mas retrasados en los grupos mas adelantados. Las ansias de obtener la recompensa que significaba ingresar al *cuadro de honor* propició la ayuda entre estudiantes.

Finalmente, con el avanzar de las clases se fueron incorporando contenidos curriculares los cuales, en este contexto, fueron recibidos de una forma lúdica, por ejemplo, se realizaron programas para la práctica de tablas de multiplicar, identificación de triángulos, cálculos de perímetros, etc.



3. **Conclusión**

La enseñanza de programación en nivel primario sin la disponibilidad de computadoras con la utilización de dispositivos digitales como los celulares, es totalmente factible y provechosa sobre todo cuando se utilicen las técnicas pedagógicas descriptas ya permiten la incorporación de conocimientos en forma lúdica permitiendo la personalización de la enseñanza graduando las tareas a fin de obtener una clase nivelada en conocimientos.

4. **Bibliografía**

- [1] Python in Education, Nicholas Tollervey, Editorial O'Reilly
- [2] Actualización en tecnología educativa – UTN.BA – Marcelo Fraire