

Orientaciones generales en programación y robótica

Contenido

1	Introducción	1
2	Orientaciones generales	1
2.1	Programación desconectada.....	1
2.2	Robótica y programación manipulativa (resultado en el robot)	2
2.3	Programación manipulativa (resultado en pantalla)	3
2.4	Programación en ordenador o dispositivos móviles (resultado en pantalla).....	3
2.5	Programación y resultado en un dispositivo físico que debemos construir (robot, placa, ...).....	4
3	Pasos en el aprendizaje en robótica y programación	4
4	Referencias.....	4

1 Introducción

El aprendizaje de los alumnos con discapacidad visual en los contenidos de programación y robótica precisan de una formación dirigida y de un acompañamiento específico en determinados momentos.

Estos contenidos se trabajan generalmente con programas muy visuales y poco accesibles con teclado. La programación se inicia con entornos gráficos de bloques. La construcción y montaje de robots y placas se realiza con elementos de reducidas dimensiones y de difícil manipulación.

El alumnado con resto visual puede trabajar la programación en estos entornos con herramientas tiflotécnicas de magnificación, en cambio, el alumnado con ceguera debe usar entornos de programación por código accesibles con el teclado.

Respecto al montaje y construcción de robots y placas todos los alumnos con discapacidad visual necesitarán adaptaciones.

2 Orientaciones generales

2.1 Programación desconectada

1. Se recomienda iniciar el proceso de aprendizaje del pensamiento computacional usando la programación desconectada. En el caso de los

alumnos con discapacidad visual les permite, alternando las funciones de robot y de programador, aprender de manera vivencial su funcionamiento y programación. Interiorizar el movimiento del robot, percibir los giros, ...

2. Cuando el alumno con discapacidad visual es el que da las instrucciones, debe acompañar al compañero robot para verificar que se llega al objetivo. O detectar, en su caso, el error.
3. Cuando el alumno con discapacidad visual hace de robot, debe recibir las instrucciones orales que deben ser simples y no muy numerosas. Tales como avanza, retrocede, giro a la derecha y giro a la izquierda. El alumno las reproduce y verifica que llega al objetivo.
4. El alumnado debe conocer antes el espacio en el que se moverá. Tomar precauciones de no tropezar, chocar con obstáculos, evitar accidentes, etc.
5. Si el alumno tiene baja visión se pueden marcar las baldosas con cinta de colores contrastados, cintas con relieve de carrocero, etc.
6. Otra forma de programación desconectada basada en el uso de fichas, tarjetas o tableros en las que se mueven personajes hacia un objetivo con flechas o instrucciones recortables, deben adaptarse de forma táctil y visual, o añadir elementos manipulativos. Por ejemplo, fichas de code.org, Scratch Jr. Tactile, etc.

2.2 Robótica y programación manipulativa (resultado en el robot)

1. Cuando el robot incorpore un tablero para la realización de la actividad deberemos añadir adaptaciones táctiles y/o visuales bien contrastadas para marcar los recorridos, puntos de salida, de llegada, objetos localizables, personajes...
2. Para facilitar la orientación en el tablero se propone reproducir a escala reducida el tablero de actividad para su reconocimiento global.
3. Cuando el robot se programe pulsando botones, si fuese necesario, deberán marcarse con elementos táctiles que indiquen la instrucción que se va a realizar (derecha, izquierda, ...).
4. Cuando la programación del robot se realice mediante fichas, cartas, tarjetas de programación, etc. Es necesario adaptarlas en relieve, braille/texturas/marcas táctiles y/o de forma visual bien contrastada.
5. Para la orientación del robot en el espacio, si es necesario, también podremos añadir marcas táctiles/visuales contrastadas que ayuden al alumno a situarlo en el tablero. Por ejemplo, poner o resaltar ojos en la parte delantera del robot.
6. Los robots deberán proporcionar feedback auditivo y visual (luces de colores) para facilitar su control y seguimiento. El alumnado debe saber que el robot ha recibido y reconocido la instrucción dada. Y durante la ejecución del programa deberá emitir feedback auditivo y visual de las acciones que está realizando, facilitando así su seguimiento.
7. En los robots con desplazamiento deberemos garantizar el acompañamiento del robot por parte del alumnado (tocándolo con

suavidad para no interferir en el movimiento del robot), facilitando recorridos en espacios y tableros reducidos, adecuados a su alcance.

8. Ejemplos de este bloque son robots del tipo BeeBot, Escornabot, Tale-Bot, Cubetto, etc.

2.3 Programación manipulativa (resultado en pantalla)

1. Cuando dispongamos de juegos de programación manipulativa de fichas o elementos que deben encajarse para realizar la programación deberemos realizar una adaptación táctil para el alumnado con ceguera.
2. El resultado en la pantalla no siempre es accesible. Deberá realizarse una adaptación de los laberintos o asegurarse que existen feedbacks auditivos. Este es el caso de juegos como Scottie go, Osmo Coding, etc.
3. Como alternativa manipulativa accesible para el alumnado con ceguera disponemos del Code Jumper, material diseñado específicamente para este alumnado.

2.4 Programación en ordenador o dispositivos móviles (resultado en pantalla)

1. En la programación por bloques, el alumnado con ceguera no puede manejar el ratón ni arrastrar los bloques utilizando gestos, debemos encontrar aplicaciones en las que se pueda programar usando el teclado. Programas como Scratch Jr., Scratch, Lego Wedo, etc. no son accesibles.
2. El alumnado con ceguera puede programar tecleando el código textual del programa (por ejemplo, Python, Arduino, Mblock, JavaScript, ...). Para ello se requieren unos conocimientos (mecanografía, lenguaje y sintaxis de programación, ...), complejos en los niveles de Primaria y Secundaria. Siempre dependerá del dominio del ordenador y de la herramienta tiflotécnica que tenga el alumnado.
3. Cuando todavía no hay dominio del lenguaje de programación se pueden preparar plantillas en las que el alumnado solo deba realizar pequeños cambios y observar el resultado. A medida que el alumnado vaya aprendiendo puede iniciarse en la edición de código.
4. El tiempo necesario para la edición de programas usando código de programación será superior al empleado arrastrando bloques. Es importante tenerlo en cuenta en el momento de diseñar la temporalización de las actividades.
5. El resultado de la programación debe ser accesible al alumnado con ceguera y baja visión. Sus actividades deben incorporar bloques o instrucciones de código con feedback auditivo (lectura de textos, audios, sonidos, ...) permitiendo al alumnado el seguimiento de los resultados del programa. Por ejemplo, los pasos del gato de Scratch, el choque con el borde de la pantalla u otro objeto, bocadillos con voz, etc.
6. En el caso de alumnado con ceguera la interacción con el programa (inicio, final, pausas, ...) debe poder realizarse usando el teclado.

2.5 Programación y resultado en un dispositivo físico que debemos construir (robot, placa, ...)

1. El orden es imprescindible para la localización de las piezas. Se recomienda la colocación en bandejas, con subdivisiones por fases de construcción, etc. Todo debidamente etiquetado para su fácil identificación.
2. Debemos proporcionar modelos físicos para copiar la construcción del robot o de las fases. Estos modelos sirven de soporte a instrucciones escritas, orales, etc. Por ejemplo, en construcciones Lego, Escornabot, etc.
3. Es imprescindible disponer de tiempo para la construcción. Si nos vemos limitados siempre podemos construir sólo una parte del robot y trabajar de forma colaborativa con compañeros.
4. Cuando se trata de montajes de placas, con componentes electrónicos y programación en Arduino, se recomienda reemplazar los montajes eléctricos con placas en las que los componentes ya estén montados, evitando las dificultades de montaje en la protoboard. En general, son componentes reducidos de tamaño y complicados de manipular. Es mejor utilizar alternativas con placas tipo BQ, Echidna, Micro: bit, conectores telefónicos, etc.
5. La programación es equiparable a lo descrito en el apartado anterior.

3 Pasos en el aprendizaje en robótica y programación

Secuenciación recomendada en el aprendizaje de conceptos de robótica y programación con alumnado con discapacidad visual:

1. El alumno es el propio robot.
2. Programación desconectada. Programación con fichas u otras alternativas, sin software. Por ejemplo, Scratch Jr. Tactile.
3. Programación con robots de botones: Beebot, Escornabot, Code & Go Mouse Robot...
4. Programación inicial con robots por fichas o elementos manipulativos: Code-oruga, BlueBot, Cubetto, Genibot, Matatalab, ...
5. Programación manipulativa con software accesible. Code Jumper.
6. Programación por código accesible: PSeInt, Python, ...
7. Placas controladoras: Micro: bit y Python. Echidna y Arduino

La construcción manipulativa de bloques tipo LEGO se realiza mediante adaptaciones con láminas en relieve e instrucciones en Braille. La programación se realiza con una adaptación manipulativa ya que el software NO es accesible.

4 Referencias

- Scratch Jr. Tactile. <https://www.scratchjrtactile.org/>

- Code Jumper. <https://codejumper.com/>
- Escornabot. <https://escornabot.com/es/index>
- Echidna. <https://echidna.es/>
- Micro:bit. <https://microbit.org/es-es/>

Ilustración 1 Ejemplos de adaptación de tableros en relieve

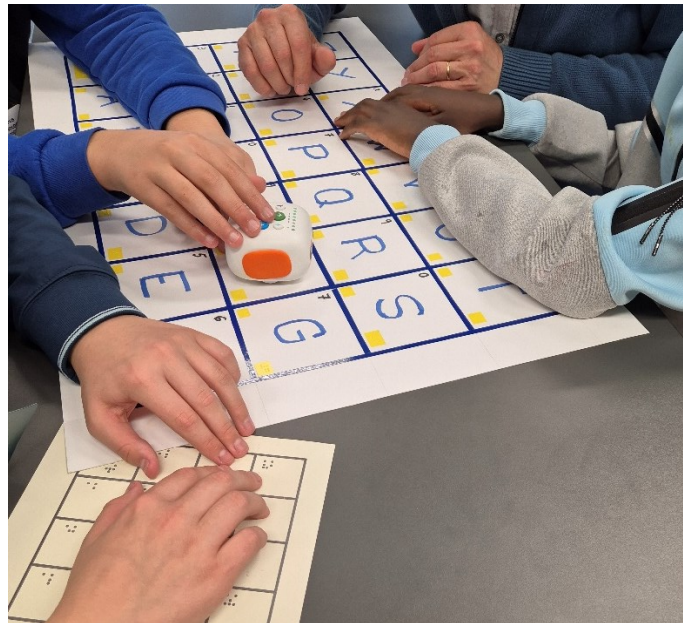


Ilustración 2 Kits de adaptación de fichas/cartas en Braille o relieve

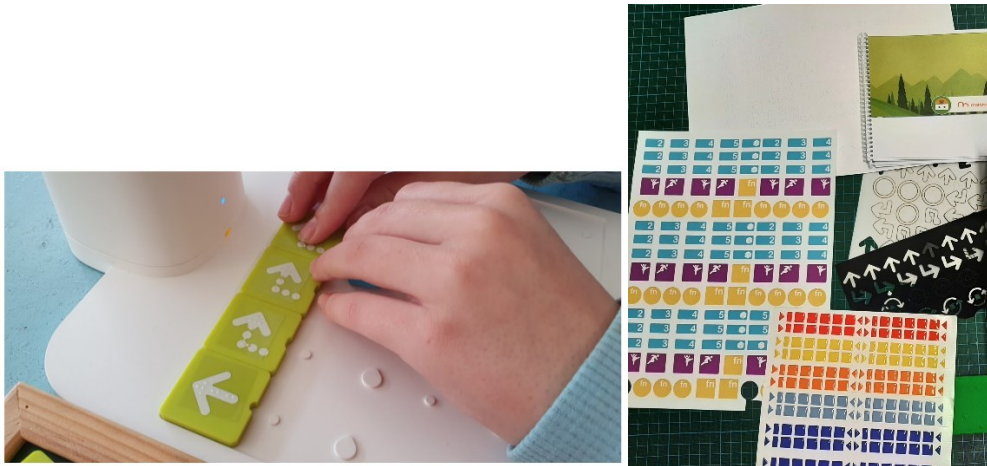


Ilustración 3 Scratch táctil Jr. Piezas manipulativas



Ilustración 4 Code Jumper. Programación manipulativa y en ordenador. Accesible con revisor de pantalla.



Ilustración 5 Programar en micro: bit con magnificación y código

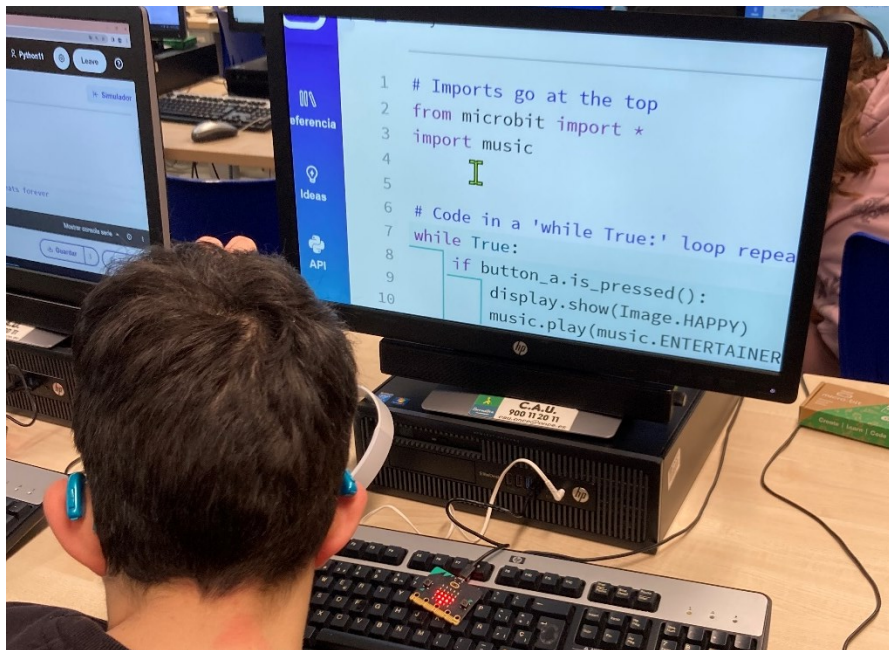


Ilustración 6 Robot con placa Micro: bit

