

SET VEINTI UNO

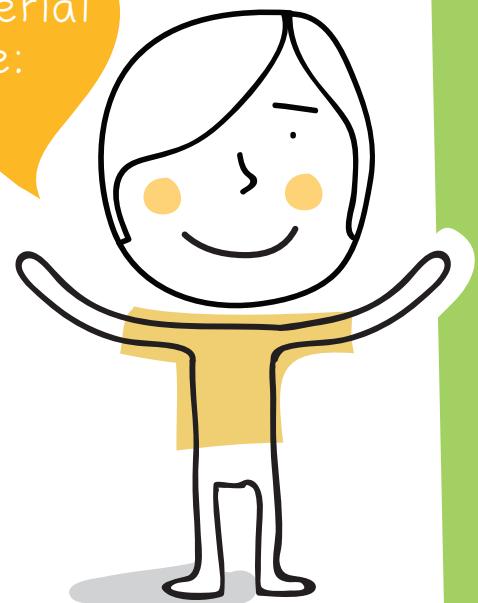
UNIDAD
MODELO



UNIDAD MODELO

- Proyecto del Cuaderno del alumno
- Cuaderno del profesor
- Acceso a la experiencia web

Este material incluye:



Navegar y Programar SET VEINTIUNO

Nos encontramos inmersos en una época de cambio apasionante. Una nueva era en la que la tecnología y la robótica se convertirán en una parte absolutamente imprescindible de nuestro día a día. Un día a día que nos trae consigo una enorme cantidad de retos y de oportunidades. Está en nuestra mano preparar a nuestros jóvenes para que sepan aprovechar esos retos.

Para ello Santillana crea *Navegar y Programar* (su nombre para el material en castellano) / *Switched on Computing* (el nombre del material en inglés), un programa que trabaja el pensamiento computacional y la alfabetización digital de los niños de Primaria, fomentando un uso responsable y ético de la tecnología.

Navegar y Programar / Switched on Computing es un programa que lleva tres años en el mercado internacional mostrando muy buenos resultados, el cual Santillana licencia y adapta para nosotros. Introduce la computación en el contexto próximo del alumno, haciéndola atractiva y cercana a la par que práctica, ya que los conocimientos son aplicables a la vida cotidiana de los alumnos desde el primer momento. Incluye un minucioso cuaderno del profesor que permite a cualquier maestro seguir paso a paso todas y cada una de las unidades y niveles que componen *Navegar y Programar / Switched on Computing*, sin necesidad de conocimientos previos o específicos de computación.

Introduciendo la computación desde tan temprana edad contribuimos, no solo a, como decíamos antes, inculcar un uso ético y respetuoso de la tecnología, sino a eliminar los estereotipos de género que las TIC llevan consigo. Trabajando la computación desde tan jóvenes, eliminamos esa idea preconcebida de que la computación es cosa de «chicos» e igualaremos los porcentajes de mujeres en puestos técnicos de empresas tecnológicas, que hoy se acerca tímidamente al 20 %.

El programa de *Navegar y Programar / Switched on Computing* consta de 6 niveles, uno por cada curso de Primaria, con 6 unidades de 6 sesiones de unos 45 minutos cada una. Los materiales están disponibles tanto en castellano como en inglés, y adaptados a las diferentes edades que comprenden la Primaria.

Además de los cuadernos del alumno y del profesor, estos materiales incluyen una serie de videos, plantillas y multitud de recursos online en la web SET VEINTIUNO, la cual no es solo un repositorio de materiales extra, sino que permitirá a los docentes la recolección de evidencias del trabajo de sus alumnos. Estas evidencias se suben al ePortfolio de la web, que permite a los profesores evaluar tanto el proceso como el resultado final. También podrá compartir su trabajo con el de otros usuarios de la comunidad SET VEINTIUNO a la vez que compartir experiencias e ideas nuevas.

Claves para entender Navegar y Programar / Switched on Computing

1

Introducción al mundo **STEAM**. Manejo de las principales herramientas digitales, primeros pasos en programación, comprensión del funcionamiento de los dispositivos digitales más comunes y creación de tecnología aplicada, implicando al arte y a la creatividad en el proceso.

2

Uso seguro, responsable y ético de las TIC, fomentando la empatía y los valores ciudadanos. A través de consejos sobre ciberseguridad y haciendo hincapié en temas como los derechos de autor y los permisos y licencias necesarios para publicar evidencias de su trabajo.

3

Desarrollo de habilidades y destrezas asociadas al **pensamiento computacional**: lógica, abstracción, descomposición de problemas, generación de algoritmos o depuración, entre otros.

4

Aprender construyendo dentro de un aprendizaje basado en proyectos. Fomento de la **experimentación como elemento motivador**. Todas las unidades están estructuradas de forma que, según se presenta la teoría, inmediatamente se pone en práctica, constituyendo un paso más hacia la consecución del reto que presenta cada unidad.

5

Desarrollo de la **perseverancia, la resiliencia y la tolerancia a la frustración** en la búsqueda de un objetivo. Se trabaja con multitud de actividades de corrección de errores de programación, normalizando los errores y la corrección de los mismos; y con actividades de crítica sobre el trabajo propio y sobre el trabajo de los compañeros.

6

Fomento de la **creatividad** dejando espacio libre a la **imaginación**, mediante enunciados del tipo: «Crea una animación en la que se cuente un chiste utilizando Scratch».

7

Eliminación de roles de género culturales asociados al mundo tecnológico mediante la introducción a temprana edad de proyectos STEAM. Aquellos niños que comienzan a generar tecnología desde muy pequeños comprobarán que no hay diferencias entre el rendimiento de niños y niñas, eliminando la idea de que el mundo tecnológico es casi exclusivamente masculino.

8

Promoción constante del **aprendizaje colaborativo**, tanto con la metodología como con la utilización de herramientas de software abierto del que tomarán materiales de otros autores (siempre respetando los derechos de autor y permisos pertinentes) y compartirán su trabajo con la comunidad online.

Cambio en el modelo de evaluación:

- Fomenta la **evaluación por evidencias**.
- Trabaja la autoevaluación y la coevaluación de los alumnos.

9

Genera **autoconfianza** en los alumnos al hacerles creadores de tecnología además de usuarios. Los niños ven que crean cosas funcionales útiles para ellos y la comunidad, lo que provoca un empoderamiento directo en ellos que se aplica a cualquier otro proyecto que quieran realizar. Ven que, si quieren, pueden.

El programa *Navegar y Programar / Switched on Computing* de SET VEINTIUNO repite la estructura de los conocimientos en cada nivel, adaptándose cada uno al momento vital de los alumnos, para consolidar un proceso práctico y eficaz en el momento de abordarlo:



Proyecto demo: 3.1. Somos programadores

En esta unidad, enmarcada dentro del nivel 3 para 8 a 9 años, los niños crearán una tira cómica animada usando los personajes que desarrollaron en unidades anteriores. La realizarán diseñando y programando la acción en el programa Scratch, aunque previamente habrán planificado todo el proceso sobre el papel.

De esta manera estarán utilizando secuencias en programas, trabajando variables y profundizando en el concepto de algoritmo.

Los alumnos utilizarán una herramienta de dibujo para crear a los personajes (si no lo hubieran hecho ya) y los fondos. Luego, programarán una animación transformando un guion gráfico en una serie de instrucciones para objetos gráficos. Trabajarán, de esta manera, no solo variables y diversos métodos de *input* y *output*, sino que estarán entrenando el razonamiento lógico para detectar y corregir errores en algoritmos y programas, habilidad absolutamente indispensable a la hora de desarrollar el pensamiento computacional.

Además, los niños serán capaces de elegir un determinado software para una tarea específica y combinarlo con otro para hacer sus tiras cómicas.

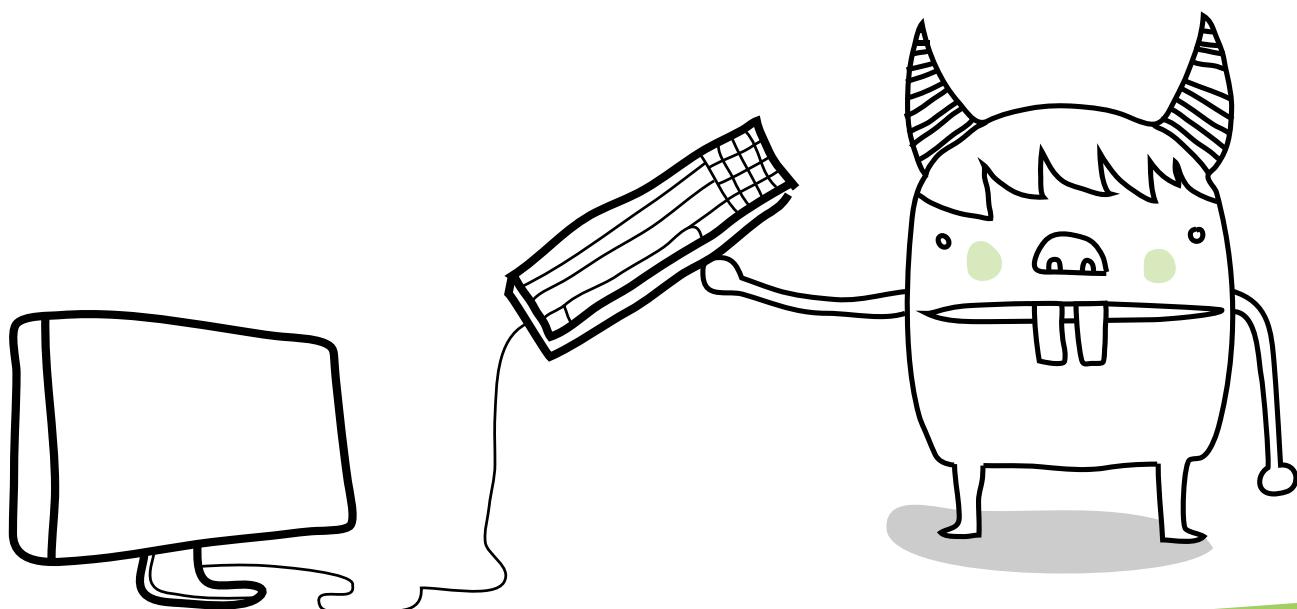
Es un momento ideal para que las presenten a la clase y trabajar el discurso en público.



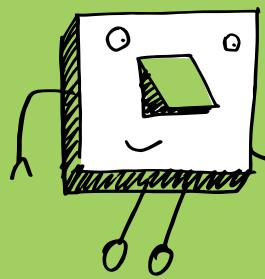
**SET
VEINTI
UNO**



NAVEGAR Y PROGRAMAR



Nivel 1
6-7 años
Cuaderno del alumno



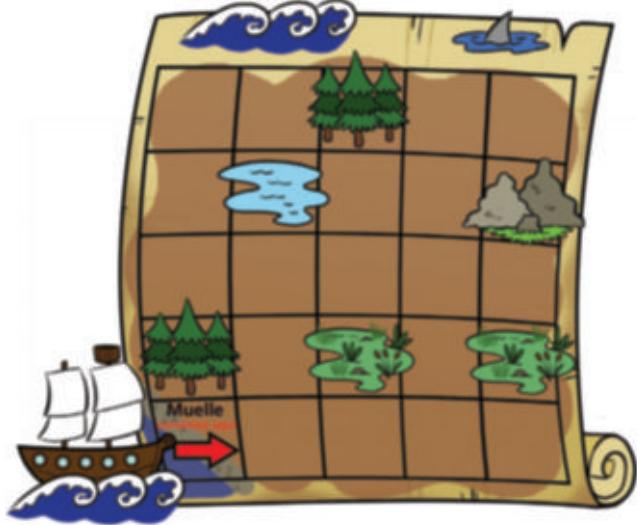
1.1

Somos buscadores de tesoros



En esta unidad:

- Aprenderás qué es un algoritmo.
- Programarás un robot para que se mueva por un mapa y encuentre el tesoro.
- Podrás predecir cómo se moverá el robot.
- Depurarás tus programas.



Palabras clave

Algoritmo: conjunto de instrucciones para solucionar un problema o realizar una tarea.

Programar: transformar las instrucciones del algoritmo a un lenguaje que el robot pueda comprender.

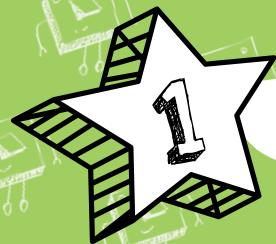
Predecir: anunciar lo que crees que ocurrirá.

Datos: información en forma de palabras o números.

Input: datos que se envían a una computadora.

Output: datos que produce una computadora.

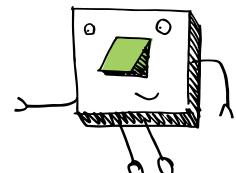
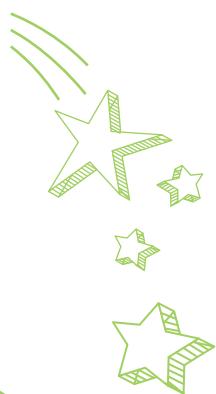
Depuración: búsqueda y corrección de errores en los programas.



Registro de un algoritmo

Aprendamos

¿Qué instrucciones tendrías que seguir para ir desde tu sitio hasta la puerta del aula?



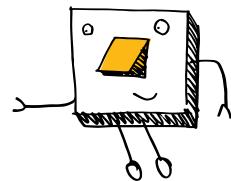
Los algoritmos son instrucciones que se utilizan para solucionar un problema o realizar una tarea.

Los algoritmos se escriben en **lenguajes** que las personas entienden, pero si queremos que una computadora realice las instrucciones del algoritmo, se lo tendremos que decir en un lenguaje que comprenda. A esto le llamamos **programar**.



Imagen de un lenguaje informático.

Probemos

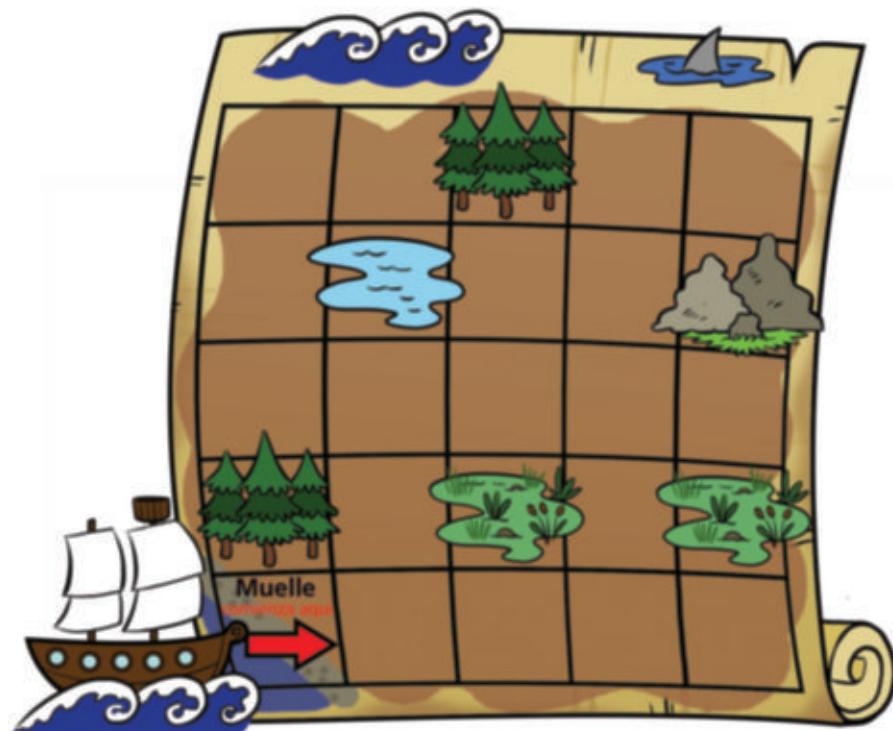


Trabajen en pareja para crear algoritmos que indiquen cómo moverse de una parte de la sala a otra.

- Uno de ustedes será el robot (la persona que escucha y sigue las instrucciones). El otro será el programador (la persona que da las instrucciones). Túrnense.
- Solo pueden usar «adelante», «atrás», «gira a la izquierda» y «gira a la derecha» como instrucciones.



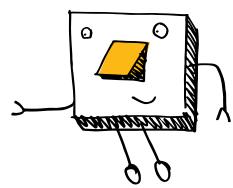
Mira el mapa del tesoro y debate con un compañero acerca de qué creen que significan los símbolos.



Probemos



Escribe un algoritmo que dirija a un robot-pirata desde el muelle hasta el lago. Usa las flechas a continuación para mostrar tu algoritmo.



- 1 ↑ = un paso hacia delante
- 1 ↓ = un paso hacia atrás
- = gira a la derecha
- ← = gira a la izquierda

Como ejemplo, fíjate en este algoritmo con instrucciones para dirigir al robot desde el muelle hasta las rocas:

3 ↑

←

3 ↑



Pide a tu compañero que siga tu algoritmo. ¿Lo dirige del muelle al lago? Si no es así, deberás depurar (corregir) tu algoritmo y asegurarte de que funciona.

2

Presentación del robot

Aprendamos

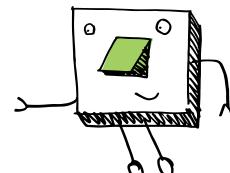
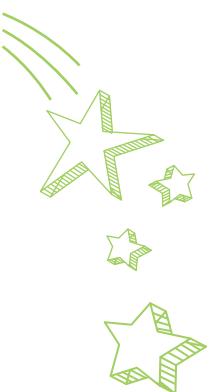
Los **juguetes** que se pueden programar, como los robots, son en realidad **computadoras**.

Los robots se pueden programar para realizar tareas o simplemente para jugar con ellos. Incluso podríamos programar uno como el del dibujo para encontrar un tesoro en un mapa, ya que tiene botones para moverse hacia delante, hacia atrás y para girar a la izquierda o a la derecha.



Al pulsar los botones le damos instrucciones al robot (**input**). El resultado (**output**) es que el robot se mueve por el suelo siguiendo esas indicaciones.

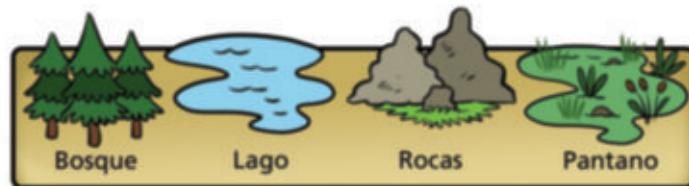
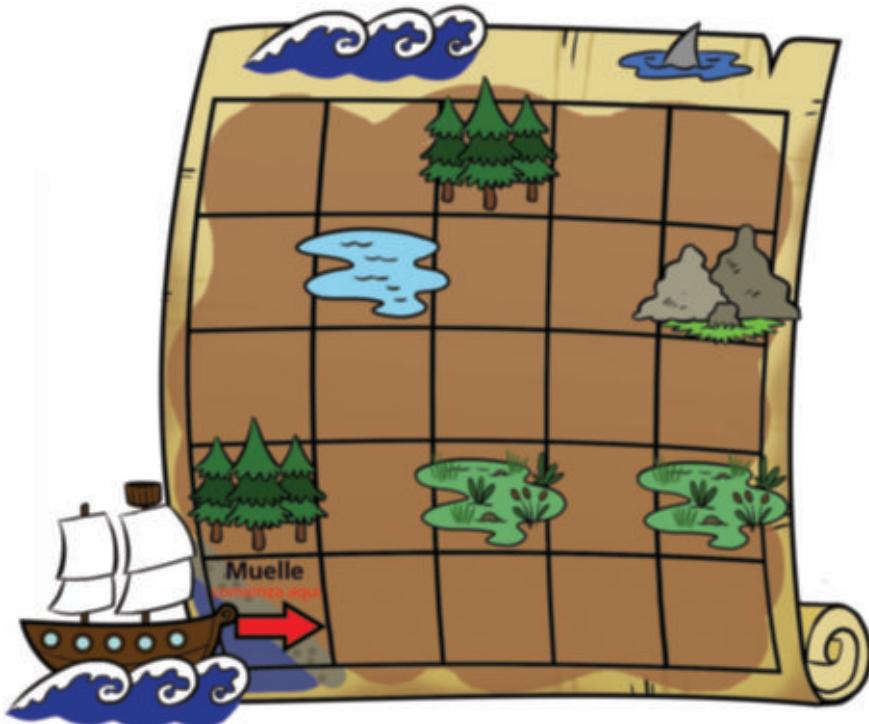
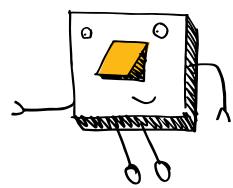
Es una manera muy divertida de buscar un tesoro escondido.



Probemos



Observa el mapa del tesoro.



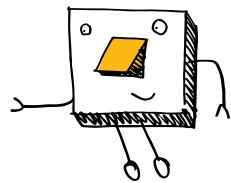
El capitán pirata Barbanegra escondió dos cofres con tesoros en la isla y escribió estas instrucciones para recordar dónde enterró el primer cofre. Léelas.

1. *Da 3 pasos hacia delante.*
2. *Gira a la izquierda.*
3. *Da 2 pasos hacia delante.*
4. *Gira a la derecha.*
5. *Da 1 paso final.*
6. *¡Comienza a excavar!*

Probemos



Trabaja en grupo para seguir las instrucciones del capitán Barbanegra moviendo una ficha por el mapa.



Escribe las instrucciones del capitán Barbanegra utilizando los símbolos de las flechas.

- | | |
|----------------|-------------------------|
| $1 \uparrow$ | = un paso hacia delante |
| $1 \downarrow$ | = un paso hacia atrás |
| \rightarrow | = gira a la derecha |
| \leftarrow | = gira a la izquierda |



Elige un lugar secreto del mapa y escribe las instrucciones, ya sea con palabras o con símbolos, para llegar hasta él. Pide a un compañero que siga las instrucciones con una ficha. ¿Logró llegar hasta tu escondite secreto?

3

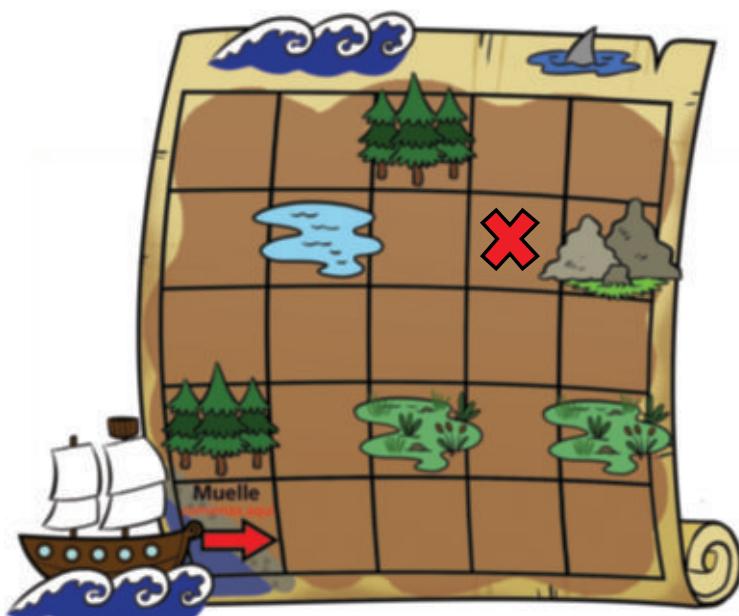
Programación del robot para encontrar el tesoro

Aprendamos

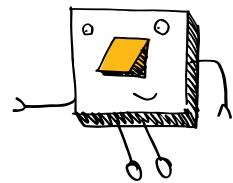
Existen varias formas de dar instrucciones para moverse por un mapa: con palabras, con flechas o con letras y números. Por ejemplo:

- 1AD puede representar «un paso hacia delante». El número puede cambiarse en función de los pasos, p. ej., 4AD significaría «cuatro pasos hacia delante».
- 1AT podría ser «un paso hacia atrás».
- D e I podría ser «gira a la derecha» y «gira a la izquierda».

Estas instrucciones se podrían utilizar para que un robot encontrara en el mapa el tesoro que enterró el pirata Barbanegra.



Probemos



Coloca una ficha al inicio del mapa del tesoro que te ha dado tu profesor. Esta representará el robot. Muévela un paso a la vez hasta llegar a la **X**.



Anota a continuación cada movimiento de la ficha.
Usa las instrucciones AD, AT, D e I.



Coloca un robot en el punto de inicio de una versión más grande del mapa. Programa de una sola vez todas las instrucciones que ya escribiste. Cuando se ejecuta el programa, ¿el robot se mueve hasta la **X** del mapa? Si no es así, ¿qué parte de tu programa debes cambiar?



Coloca el robot en otra casilla del mapa. Prográmalos para que se mueva hasta la **X** nuevamente.

4 Depuración

Aprendamos

A veces los robots no se comportan como esperábamos porque hay errores en su programación. Para arreglarlo, hay que **depurar**, que significa comprobar las instrucciones del programa y corregir los errores. La depuración es una parte muy importante de la programación.

Una forma útil de depurar los programas es leer las instrucciones y predecir lo que crees que sucederá. Pedir a un compañero que revise las instrucciones y explicárselas también te ayudará a encontrar errores.

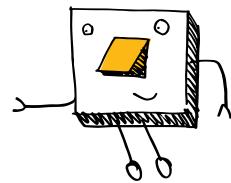


Hay que observar detenidamente los programas para encontrar los errores que esconden.

Probemos



Lee con un compañero las siguientes instrucciones para moverte por el mapa del tesoro.



¿De este modo llegas desde el muelle al lago?

1. 3AD
2. I
3. 2AD
4. I

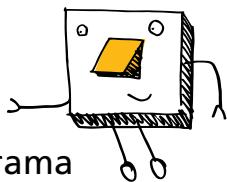
¿De este modo llegas desde el muelle a las rocas?

1. 2AD
2. 1AT
3. I
4. 2AD



Predice qué sucederá si estas instrucciones se programan en un robot. Conversa sobre este tema con un compañero.

Probemos



Sugírele a tu compañero formas de arreglar el programa y escríbelas a continuación.



Trata de ingresar el algoritmo corregido en un robot. ¿Necesitas cambiar algo más? Conversa sobre este tema con tu compañero.



Actividad

- ★ Programa dos robots para que bailen.
- ★ Piensa en cómo se moverán. ¿Quieres que se muevan juntos? ¿Prefieres que giren en diferentes direcciones?



Mi progreso



Nombre:



Clase:

¡Inserta aquí una imagen
de tu progreso!

1.1 Somos buscadores de tesoros

Sé seguir instrucciones.



Sé registrar un conjunto de instrucciones.



Soy capaz de programar un juguete.



Puedo dar instrucciones a otros.



Entiendo lo que significan *input*, programa y *output* en
un robot de juguete.



Puedo dar ejemplos de *input*, programa y *output*.



Soy capaz de crear un programa.



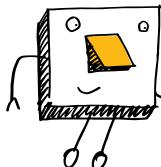
Sé cómo detectar y corregir errores en un programa (depurar).



Puedo predecir a dónde llevará un conjunto de instrucciones
a un juguete o a una persona.



Puedo buscar formas de mejorar un programa.



Reflexionemos

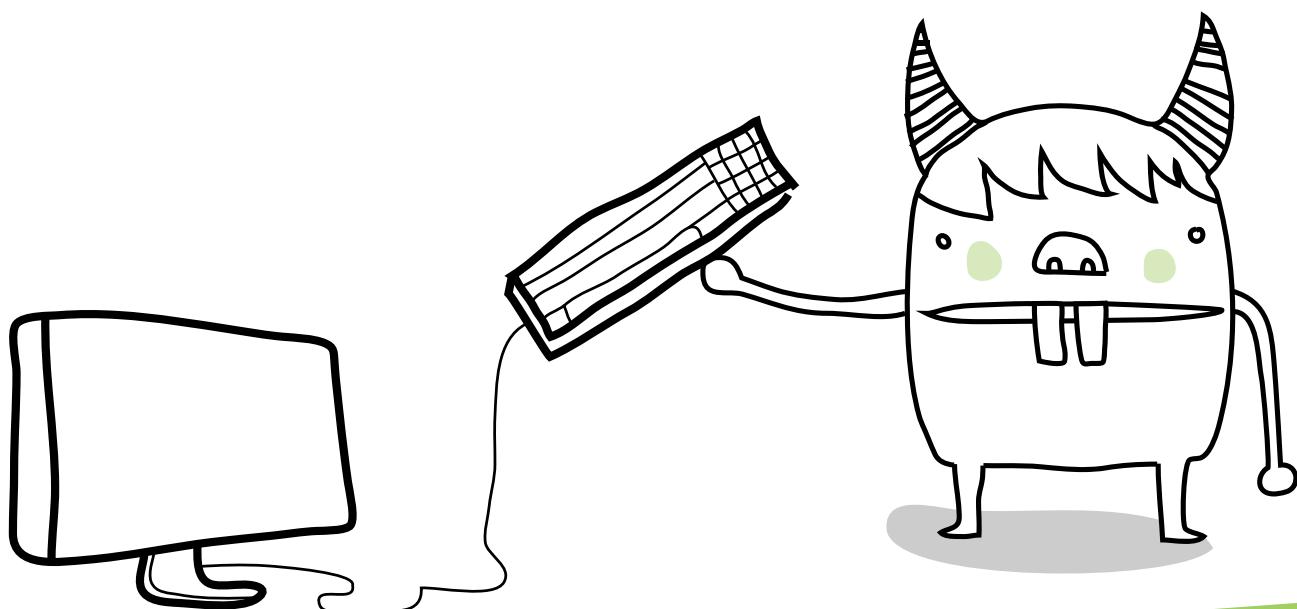
Soy bueno para ...

La próxima vez ...

**SET
VEINTI
UNO**



NAVEGAR Y PROGRAMAR



Nivel 1
6-7 años
Cuaderno del profesor



Somos buscadores de tesoros

Uso de juguetes programables

1

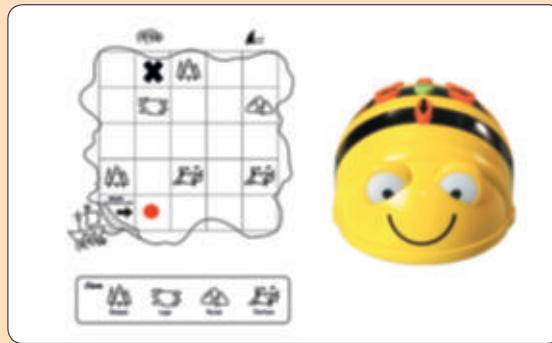
Acerca de esta unidad

Software: Interfaz de programación para un juguete programable

Apps: Aplicación Bee-Bot y Daisy the Dinosaur

Hardware: Juguete programable, como Bee-Bot o Roamer Too. Para el primer paso se necesitan grabadoras de audio (el teléfono puede servir)

Resultado: Secuencia de instrucciones que harán mover un juguete programable por una ruta determinada



RESUMEN DE LA UNIDAD

En esta unidad, los niños programarán un juguete para que se mueva por el mapa y encuentre el tesoro enterrado. Comenzarán pensando en algoritmos para las rutas y luego los insertarán como programas almacenados para el robot. Predecirán cómo se moverá el robot y depurarán los programas.

RELACIÓN CON EL PLAN DE ESTUDIOS

Estándares de aprendizaje evaluables

- Comprender qué son los algoritmos y cómo se implementan como programas en dispositivos digitales. Entender que los programas se ejecutan con instrucciones precisas que no son ambiguas.
- Crear y depurar programas simples.
- Usar el razonamiento lógico para predecir el comportamiento de programas simples.
- Reconocer los usos comunes de la tecnología de la información fuera de la escuela.

Vínculos sugeridos

- **Geografía:** Los niños utilizan un idioma geográfico y pueden usar y crear sus propios mapas.
- **Matemáticas:** Los niños reconocen los movimientos en línea recta (traslaciones) y las rotaciones y las combinan de forma sencilla.
- **Historia:** Los niños averiguan cosas sobre el pasado.
- **Lengua:** Los niños proporcionan instrucciones claras para moverse por el mapa.

DESCRIPCIÓN DEL PROGRAMA DE COMPUTACIÓN

- Al crear sus instrucciones, los alumnos crean algoritmos, es decir, instrucciones paso a paso para alcanzar una determinada meta. Programar implica convertir esas instrucciones en un lenguaje formal que comprenda la computadora. En este caso, una serie de botones que hay que presionar

en un robot de juguete. El robot simplemente sigue las instrucciones que se le dan.

- A veces, los programas de los alumnos no resultan como lo planearon. En estos casos, deben corregir (depurar) sus programas para reparar los errores. Los alumnos deben tener una idea clara de lo que hará el programa usando un razonamiento lógico para predecir lo que sucederá a partir de las instrucciones.

OBJETIVOS DE APRENDIZAJE

Esta unidad permitirá a los niños:

- Entender que un juguete programable puede controlarse al ingresar una secuencia de instrucciones.
- Desarrollar y registrar secuencias de instrucciones como un algoritmo.
- Programar los juguetes para que sigan los algoritmos.
- Depurar sus programas.
- Predecir cómo funcionan sus programas.

La guía de evaluación en la página 20 te ayudará a decidir si los niños han alcanzado estas expectativas.

OTRO ENFOQUE

- Los niños pueden ordenar a un juguete programable que realice un recorrido siguiendo el plano de la escuela.
- Los niños pueden hacer que el robot sea un repartidor y pedirle que se mueva entre puntos de referencia de su entorno.
- Los niños pueden usar el juguete programable para contar y hacer ejercicios de suma y resta.
- Muchos juguetes programables tienen equivalentes en pantalla o también puedes crear tu propio simulador con Scratch 2.0.

→ 2 Preparación

¿QUÉ HACER?

- Lee la sección «Pasos fundamentales» de «Desarrollo de la tarea».
- Decide qué juguete programable utilizarás. Si es posible, el juguete programable debe limitarse a hacer giros de 90 grados únicamente (o varios giros).
- Piensa en tus alumnos y valora si algún contenido de ampliación, propuestos en las páginas 14 a la 19, puede ayudarte a motivar a los niños que más destacan. Considera también algunas de las sugerencias de «Integración» para ayudar a los niños con necesidades educativas especiales.
¿Has pensado cómo podría ayudarlos un asistente de enseñanza si lo hubiera en tu escuela?
- Verifica que las baterías de los juguetes programables estén cargadas. La tapa de la batería debe estar siempre atornillada en su lugar cuando los robots estén en uso.

¿QUÉ NECESITAS?

- Copias impresas del mapa del tesoro (consulta setveintiuno.com).
- El mapa del tesoro en papel o cartón que es igual al mapa que hay en la web de SET VEINTIUNO. El tamaño de la cuadrícula del mapa debe ser equivalente a la distancia que avanza el juguete programable al dar un paso. Por ejemplo, si Bee-Bot y Roamer Too se mueven en pasos de 15 cm, la cuadrícula del mapa debe medir 75 × 75 cm. En la web se proporcionan una hoja con puntos de referencia y una X. Los puntos de referencia se pueden pegar en el mapa y la X se puede colocar en el cartón.
- Lápices.
- Juguetes programables y el software relacionado.
- Computadoras de escritorio, portátiles o tablets con el software correspondiente instalado para las actividades de ampliación.
- Fichas (consulta el paso 3 de «Desarrollo de la tarea», página 16).
- Una grabadora de audio (o la aplicación Voice Memo en tu teléfono móvil).



RECURSOS

- Mapa del tesoro.
- Archivo de Scratch de la isla del tesoro.
- Plantilla de símbolos.
- Instrucciones de Barbanegra (1 y 2).
- Un programa que necesite una depuración.
- Póster de la unidad: ¿Puedes depurar el programa?



SEGURIDAD ONLINE

- Si los niños acceden a Internet, asegúrate de que el acceso a material inapropiado esté bloqueado por los filtros.
- Si los niños son filmados o se filman unos a otros trabajando con los robots, asegúrate de obtener todos los permisos necesarios.



INTEGRACIÓN

- Algunos niños quizás necesiten programar su juguete con una cantidad menor de secuencias de instrucciones o con secuencias menos complicadas. Si este es el caso, considera la ubicación del tesoro en el paso 3 de «Desarrollo de la tarea» en la página 16.
- Es posible que algunos niños tengan dificultades para usar los botones para programar el robot. Considera poner a los alumnos en parejas o en grupos para que todos puedan participar.



ENLACES DE INTERÉS

Software y herramientas

- Bee-Bots: www.tts-group.co.uk/shops/tts/Products/PD1723538/Bee-Bot-Floor-Robot.
- Roamer Too: www.valiant-technology.com/uk/pages/roamertoohome.php?cat=8&8. El teclado para niños pequeños servirá para esta unidad: www.valiant-technology.com/us/pages/foundation.php.
- La aplicación Bee-Bot (<https://itunes.apple.com/gb/app/bee-bot/id500131639>) es gratuita. Si bien es más avanzada, la aplicación Daisy the Dinosaur cuenta con una buena plataforma de introducción a la programación: <https://itunes.apple.com/gb/app/daisy-the-dinosaur/id490514278>.
- Ve los simuladores en pantalla para Bee-Bots (www.tts-group.co.uk/shops/tts/Products/PD1722395/Focus-on-Bee-Bot-Lesson-Activities-1/?rguid=1c6eb9f8-7fe8-4b52-b244-891972ce6b40) y el Roamer original (www.valiant-technology.com/uk/pages/roamer_rworld.php?cat=1id0).
- La función *Crear un bloque* en Scratch 2.0 te permite crear tu propio simulador de un robot. Consulta <https://scratch.mit.edu/projects/121768826/>.

Tutoriales online

- Bee-Bot: www.youtube.com/watch?v=52ZuenJIFyE.
- Roamer Too: <http://vimeo.com/49152214>.

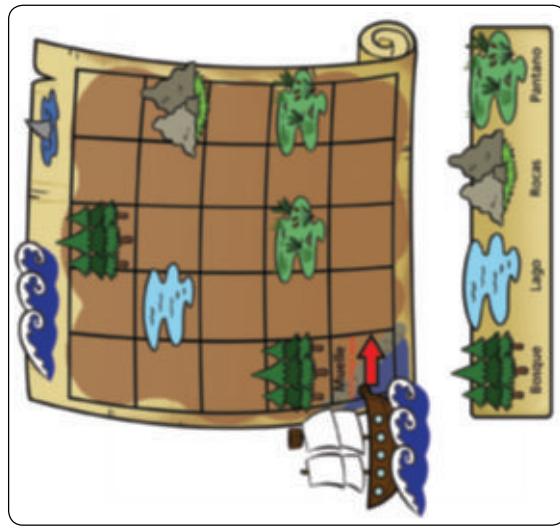
3

Desarrollo de la tarea: Somos buscadores de tesoros

Pasos fundamentales

Paso 1: Búsqueda del tesoro

POSIBLE RESULTADO:



- Lleva a los alumnos a un lugar amplio, como el pasillo o el patio de la escuela. Coloca el tesoro (puede ser una caja o una bolsa) en un lugar y reúne a los alumnos en otro punto.
- Di a los alumnos que decidan qué indicaciones necesitarían seguir para llegar a donde está el tesoro. Recuérdales que solo pueden usar «adelante», «atrás», «gira a la izquierda» y «gira a la derecha» como instrucciones. Si fuera necesario, refuerza los conceptos de izquierda y derecha.
- Un alumno debe ser el robot-pirata. Otro alumno (el programador) debe leer las instrucciones en voz alta al robot-pirata, quien deberá seguirlas al pie de la letra. ¿Consiguieron llegar hasta el tesoro? ¿El programador cambió las instrucciones a medida que veía lo que sucedía o se apegó al plan?
- Repite el ejercicio, ya sea moviendo el tesoro o a los alumnos de lugar. Esta vez, utiliza una grabadora de audio para grabar las instrucciones del programador. Pide a los demás alumnos que predigan a dónde irá el robot-pirata (incluso pueden quedarse parados en el lugar a donde creen que se dirigirá) y, después, reproduce las instrucciones para comprobarlo. Si el tiempo lo permite, los alumnos pueden repetir el ejercicio en grupos pequeños.

- Los alumnos pueden registrar sus instrucciones con palabras o imágenes.
- Los alumnos pueden repetir el mismo ejercicio en sus casas, ya sea programando ellos mismos o dejando la programación a cargo de sus padres o tutores.

Software: Interfaz de programación para un juguete programable **Apps:** Bee-Bot y Daisy the Dinosaur **Hardware:** Juguete programable, como Bee-Bot o Roamer **Resultado:** Una secuencia de instrucciones que harán que un juguete programable se mueva por una ruta determinada

Cuaderno del clúmne



REGISTRO DE UN ALGORITMO

Aprendamos

Probemos

Ampliación

EN LA ESCUELA

- Los alumnos pueden registrar sus instrucciones con palabras o imágenes.

EN CASA

- Los alumnos pueden repetir el mismo ejercicio en sus casas, ya sea programando ellos mismos o dejando la programación a cargo de sus padres o tutores.

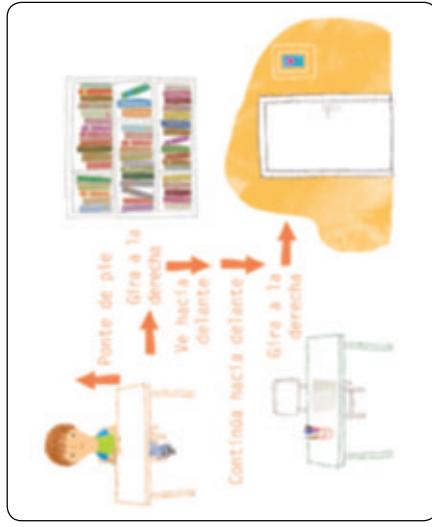
Paso 2: Registro de un algoritmo

RECURSOS

- Mapa del tesoro
- Plantilla de símbolos

SETE
VENTI
UNO

POSSIBLE RESULTADO:



REGISTRO DE UN ALGORITMO

Probemos

EN LA ESCUELA

- Reúne a la clase alrededor del mapa grande (que está en papel o en cartón). Coloca un juguete pequeño en el mapa y explica a los niños que deben darle instrucciones a la figura para que se mueva de un lugar del mapa a otro, p. ej., del lago hasta los árboles. Dales algunas ideas sobre las indicaciones que pueden dar, p. ej., adelante, atrás, girar a la izquierda y girar a la derecha.
- Coloca la figura de un juguete en el muelle, en el mapa grande, y diles a los niños a dónde deben ir. Todos en clase deben darle instrucciones a la figura para que se mueva, un paso a la vez, hasta su destino. A medida que la figura se mueva, escribe cada instrucción en la pizarra interactiva. Pueden acordar en clase un formato común para hacerlo, ya sea con símbolos, palabras o ambos. Por ejemplo, avanzar un paso podría registrarse como: ↑ 1, Adelante 1 o 1AD.
- Una vez que la figura llegue a su destino, vuelve a colocarla en el muelle y sigue las instrucciones recién registradas. Asegúrate de que los niños entiendan que usar esas indicaciones siempre implicará que, comenzando desde el muelle en dirección este, la figura termine siempre en el mismo lugar.
- Pide a los alumnos que tomen nota de las instrucciones necesarias para mover la figura desde el muelle hacia otro lugar, pero esta vez tomando nota de todas las indicaciones a la vez en lugar de una por una. Explica que deben pensar cuidadosamente (usar un razonamiento lógico) para averiguar hacia dónde irá el juguete y cómo quedará con cada paso. Pide a los niños que sigan las instrucciones que escribieron. ¿Funcionaron? Si no, los alumnos deberán corregirlas (depurarlas) y comenzar nuevamente.

- Los niños pueden explorar las instrucciones dadas por Google Maps para moverse entre dos lugares, como la escuela y la biblioteca local: www.google.com/maps.

EN CASA

- Alienta a los alumnos a que hablen con sus padres o tutores sobre indicaciones que reciben habitualmente en el hogar, por ejemplo, cómo prepararse para ir a la escuela.

- Proyecta el archivo en PDF del mapa del tesoro (setveintiuno.com) en la pizarra interactiva. Habla con ellos sobre lo que ven. ¿Qué creen que representan los símbolos que hay en el mapa?

Pasos fundamentales

Paso 3: Presentación del robot

POSIBLE RESULTADO:



- Preséntale el juguete programable elegido a los niños. Da tiempo para que lo prueben y descubran cómo funciona, especialmente si es un juguete que no conocen. Pide a los alumnos que comparten con la clase lo que han descubierto. Corrige las nociiones incorrectas. Muéstrales qué botones presionar para que el juguete se mueva un paso a la vez, si es que no lo descubrieron por sus propios medios. Dales tiempo a los niños para que se turnen (ya sea en grupos o en parejas).
- Explica que el juguete es una computadora: acepta *inputs* y produce *outputs* conforme a un programa almacenado. Pide a los niños que piensen en otras tecnologías que también acepten *inputs*, almacenen programas y produzcan *outputs*. Corrige las nociiones incorrectas.
- Muéstrales cómo se puede programar al robot interactivamente (es decir, un paso a la vez) para moverse por el mapa de la isla. Pide a los alumnos que utilicen una programación paso a paso para moverlo de un lugar a otro.
- Reparte a los alumnos en grupos y pídeles que piensen en desafíos de programación similares. Diles que los realicen, de forma que muevan el robot de un lugar a otro.
- Muestra a los alumnos instrucciones para moverse de un lugar a otro y pídeles que predigan dónde terminará el robot. ¿Lo hicieron bien? ¿Cómo lo resolvieron? En grupo, los alumnos pueden pensar en desafíos de programación similares y realizarlos. Deberán dar instrucciones y predecir la ubicación en la que terminará el robot cuando se programen estas instrucciones (una a la vez).

Cuaderno del alumno



PRESENTACIÓN DEL ROBOT

Aprendamos

Ampliación

EN LA ESCUELA

- Los alumnos pueden escribir sus propias instrucciones.

EN CASA

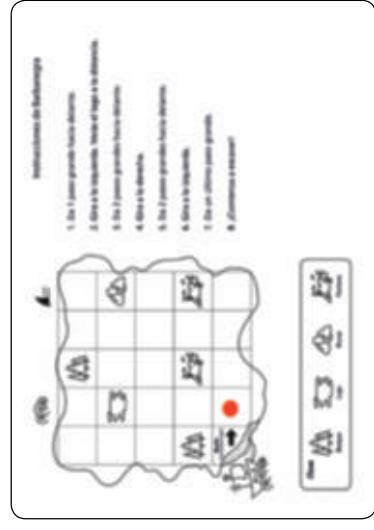
- Los alumnos deben buscar otros dispositivos en sus casas que acepten *inputs*, almacenen programas y produzcan *outputs*, como consolas de juegos, teléfonos inteligentes, tablets, hornos microondas y lavarropas.

Paso 4: Introducción a la programación

RECURSOS

- Mapa del tesoro
- Instrucciones de Barbanegra (1 y 2)

POSIBLE RESULTADO:



PRESENTACIÓN DEL ROBOT

Probemos

EN LA ESCUELA

- Pide a los alumnos que piensen si los algoritmos que utilizan son la mejor manera de llegar al destino.

EN CASA

- Explicales que un juguete programable será el buscador del tesoro y que si lo programan con las instrucciones de Barbanegra, llegará hasta él. Coloca el juguete programable en el mueble y pide a uno o más alumnos que ingresen las instrucciones de Barbanegra como una secuencia completa. Pide a los niños que piensen en dónde terminará el juguete. Muéstralas si han acertado. Pregúntales cómo hicieron sus predicciones.
- Repite el procedimiento con otras instrucciones o con un punto de inicio diferente y pide de nuevo a los alumnos que predigan dónde terminará el juguete cuando se lo programe. Uno o más niños deben programar las instrucciones para que la clase pueda comprobar sus predicciones.
- Trabajando en grupos, los niños pueden escribir una secuencia de instrucciones para llegar a un punto que elijan del mapa. Después, otros niños pueden predecir y probar a dónde llegará el juguete.
- Dales tiempo para que experimenten un poco más con el juguete. Diles que piensen qué otros programas más complejos podrían programarse en el robot y, en cada caso, cómo respondería el robot.

Pasos fundamentales

Paso 5: Programación del robot para encontrar el tesoro

RECURSOS

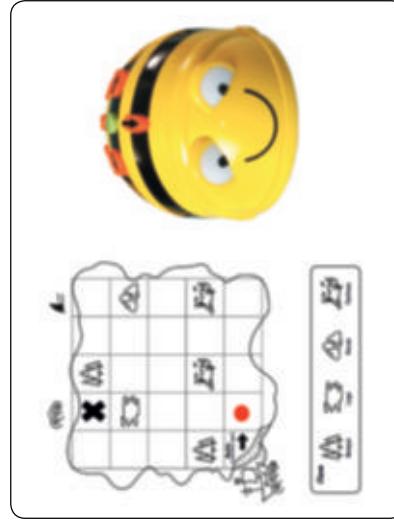
Mapa del tesoro



<https://scratch.mit.edu/projects/121770070/>



POSIBLE RESULTADO:



Cuaderno del alumno



Aprendamos

Probemos

Ampliación

EN LA ESCUELA

- Divide a la clase en grupos. Entrega a cada grupo una versión impresa del mapa del tesoro en el que hayas marcado una X (que representa otro tesoro enterrado). Puedes marcar la X en diferentes lugares para cada grupo, según sus capacidades.
- Entrega a cada grupo una ficha que represente un juguete programable. Pide a cada grupo que coloque la ficha en el mueble y que la mueva un paso a la vez hasta llegar a la X, tomando nota de cada movimiento de la ficha (con lápiz y papel o con una computadora portátil o de escritorio, según corresponda).
- Los grupos deberán ver si encuentran el tesoro usando el juguete programable en el mapa del tesoro grande. Primero, pídeles que coloquen el juguete en el mueble y la X en el cuadrado del mapa grande impreso. Pídeles que ingresen la secuencia de instrucciones que hayan anotado y que luego prueben si las instrucciones dirigen el juguete hasta el tesoro.
- Cambia la ubicación de inicio o la ubicación del tesoro a medida que los grupos resuelvan cada problema o alíéntalos a plantearse retos de programación entre ellos, como moverse de un lugar a otro del mapa.
- Puedes incluir mayor complejidad pidiéndoles instrucciones para visitar dos o más lugares, o para evitar alguna parte en especial del mapa. En cada caso, pide a los alumnos que escriban su algoritmo y que programen el robot. Todo el grupo debe hacer predicciones y probarlas.

EN CASA

- Los niños podrían usar software relacionado con el juguete programable o utilizar los bloques personalizados de Scratch para programarlo y moverlo por la pantalla (consulta el archivo de Scratch en «Recursos»).
- Si los alumnos tienen acceso a tablets o teléfonos con iOS en sus casas, pueden jugar con la aplicación gratuita Bee-Bot.

Paso 6: Depuración

RECURSOS

- Un programa que necesita una depuración

- Póster de la unidad: ¿Puedes depurar el programa?

- Grace Hopper: https://es.wikipedia.org/wiki/Grace_Murray_Hopper

POSIBLE RESULTADO:



- Muéstrales de nuevo a los alumnos el mapa y las instrucciones erróneas (consulta «Recursos»). Explica que el algoritmo tenía como finalidad mover el robot de un punto a otro, pero que crees que puede estar equivocado. Pide a los alumnos que predigan qué pasaría si se escribiera como un programa. ¿Dónde terminaría el robot? ¿Se les ocurre cómo arreglar el programa? Explica que a esto se lo llama «depurar». Cuéntales la historia de Grace Hopper, a quien se le atribuye el invento de la palabra *debug* (depurar en inglés) por quitar literalmente un insecto (*bug*) de una computadora para que funcionara (consulta la sección «Recursos»).
- Prueba una o más correcciones sugeridas por los alumnos para el algoritmo y programalo en el robot. ¿Funcionó correctamente? ¿Se necesita hacer más cambios?
- Pon a los niños el reto de buscar y corregir errores en los programas del robot de los otros, ya sea por razonamiento lógico solamente o probando los programas en el robot.
- En una reunión con toda la clase, pregunta a los alumnos por las estrategias que han adoptado para verificar y corregir los algoritmos y los programas de los otros. Pídeles que piensen en casos en los que es realmente importante que los informáticos y los programadores se aseguren de que no haya errores en los algoritmos o en los programas.
- Por último, en la sección «Actividad», los alumnos ordenarán a dos juguetes programables para que bailen.



Aprendamos

Probemos

Actividad

EN LA ESCUELA

- Los alumnos pueden hacer una carrera para conseguir el tesoro: desde un punto a otro del mapa, programando el robot para que lo haga en un tiempo determinado, en colaboración en un mismo mapa o jugando en paralelo.

EN CASA

- Pide a los alumnos que hablen con sus padres o tutores sobre ejemplos de programas (que usan en sus computadoras o en otros dispositivos) que no funcionan exactamente como deberían. ¿Se trata de errores?

4

Guía de evaluación

Usa esta página para valorar el conocimiento y las habilidades de computación de los niños. La aplicación de estos criterios de evaluación es compatible con aquellos que se apliquen en tu escuela para evaluar el trabajo de los alumnos.

TODOS LOS NIÑOS DEBERÍAN PODER:

- Seguir instrucciones para moverse por un espacio amplio.
- Registrar instrucciones para un juguete programable.
- Programar un juguete para que se mueva dándole una instrucción a la vez.
- Programar un juguete para que se mueva al darle indicaciones.

LA MAYORÍA DE LOS NIÑOS PODRÁN:

- Dar instrucciones para moverse por un espacio amplio.
- Entender lo que significan *input*, programa y *output* en el contexto de un robot de juguete.
- Crear un programa para mover un juguete a un lugar en particular.
- Depurar un programa.

ALGUNOS NIÑOS PODRÁN:

- Predecir a dónde llevarán las instrucciones a un alumno que se mueve en un espacio amplio.
- Predecir en dónde terminará un juguete a partir de determinadas instrucciones.
- Entender lo que significan *input*, programa y *output* en contextos más generales.
- Buscar maneras de que un programa sea más eficiente.

AVANCE

Las siguientes unidades permitirán a los niños continuar desarrollando su conocimiento y sus habilidades.

- *Unidad 2.1: Somos astronautas*
- *Unidad 2.2: Somos probadores de juegos*

RELACIÓN CON LOS ESTÁNDARES DE APRENDIZAJE EVALUABLES

- Comprender qué son los algoritmos.
- Crear programas simples.
- Los programas se ejecutan con instrucciones precisas que no son ambiguas.
- Crear programas simples.

- Comprender qué son los algoritmos.
- Los programas se ejecutan con instrucciones precisas que no son ambiguas.
- Crear programas simples.
- Depurar programas simples.

- Usar el razonamiento lógico.
- Usar el razonamiento lógico para predecir el comportamiento de programas simples.
- Reconocer los usos comunes de la tecnología de la información fuera de la escuela.
- Depurar programas simples.



5 Ideas para el aula

¡Sugerencias prácticas para animar esta unidad!



MUESTRA Y ACTIVIDADES

- Una pantalla que muestre a los niños trabajando con el juguete programable junto a los mapas impresos en hojas A4 y las instrucciones escritas que prepararon.
- Los niños pueden investigar qué tecnología utilizan los arqueólogos para buscar tesoros en la actualidad, como dispositivos con GPS para localizar la ubicación exacta del sitio de interés, un georadar o detectores de metales para encontrar artefactos.
- Se puede usar una bandeja grande de arena para crear una isla desierta con un tesoro enterrado, y usar un detector de metales para encontrarlo.
- Los niños pueden escuchar o aprender una canción pirata: <http://learnenglishkids.britishcouncil.org/en/songs/the-pirates-song>.
- Se puede crear un área para hacer una dramatización y comparar la vida en un crucero con la de un barco pirata. Se podría explorar la tecnología existente en las embarcaciones modernas y compararla con otras más antiguas: GPS vs. brújula, o teléfono satelital vs. código morse.



VISITAS

- Se podría visitar un museo donde haya una exposición de dispositivos de uso cotidiano que respondan a instrucciones.
- Visita un museo donde haya una exposición sobre piratas.
- Muchas fábricas utilizan robots como parte de su proceso de fabricación. También podría visitarse el departamento de Ciencias de la Computación de una universidad cercana.
- Los alumnos pueden visitar una juguetería local para investigar acerca de otros robots de juguete.



LIBROS

- Brown, H. *The Robot Book*. (Accord Publishing, 2013)
- Lloyd Jones, R. *Pirate Ships (See inside)*. (Usborne Publishing Ltd, 2007)
- Lydon, A. *Let's Go With Bee-Bot Book*. (TTS, 2008)
- Montgomery, R.A. *Your Very Own Robot*. (Chooseco, 2007)
- Murphy, G. *Robots, Chips and Techno Stuff*. (Macmillan Children's Books, 2011)
- Punter, R. *Stories of Pirates*. (Usborne Publishing Ltd, 2007)
- Storms, P. *The Pirate and the Penguin*. (Owlkids, 2009)



6 Contenido extra

Una vez que hayas finalizado, es posible que quieras ampliar el proyecto de las siguientes maneras.

- Explora otras maneras de usar lo que han aprendido los niños en esta unidad. Los juguetes programables permiten a los alumnos practicar con las distancias y las medidas (cuántos centímetros ha viajado el juguete, por ejemplo) o plantear y resolver problemas matemáticos.
- Los niños pueden colocar lápices o bolígrafos en los juguetes programables para hacer que dibujen formas sobre el papel.
- Los niños programan en la *Unidad 2.1: Somos astronautas*.



Notas



Notes



5 Classroom ideas

Practical suggestions to bring this unit alive!



DISPLAYS AND ACTIVITIES

- A display showing the children working with the programmable toy alongside their A4 hard-copy maps, and the written instructions they made.
- The children could find out about the technology archaeologists use to find modern-day treasure, such as GPS devices to locate the exact position of a site of interest, and ground-penetrating radar or metal detectors to find artefacts.
- The children could use a large sand tray to create a desert island with buried treasure, which they could use metal detectors to find.
- The children could listen to or learn a pirate song: <http://learnenglishkids.britishcouncil.org/en/songs/the-pirates-song>.
- The children could create a cruise ship role-play area and compare life on this with that of a fantasy pirate ship. In particular, they could explore the technology that is now available on modern vessels, such as GPS tracking versus compass, satellite phone versus Morse code.



VISITS

- Visit a museum with an exhibition of everyday devices that respond to instructions.
- Visit a museum with a pirate exhibition.
- Local factories may use robots as part of their manufacturing process, or a local university might welcome a visit to its computer science department.
- Pupils could visit a local toy store to investigate other robotic toys.



BOOKS

- Brown, H. *The Robot Book*.
(Accord Publishing, 2013)
- Lloyd Jones, R. *Pirate Ships (See Inside)*.
(Usborne Publishing Ltd, 2007)
- Lydon, A. *Let's Go With Bee-Bot Book*.
(TTS, 2008)
- Montgomery, R.A. *Your Very Own Robot*.
(Chooseco, 2007)
- Murphy, G. *Robots, Chips and Techno Stuff*.
(Macmillan Children's Books, 2011)
- Punter, R. *Stories of Pirates*.
(Usborne Publishing Ltd, 2007)
- Storms, P. *The Pirate and the Penguin*.
(Owlkids, 2009)



6 Taking it further

When you've finished, you might want to extend the project in the following ways.

- Explore other ways of using the techniques the children have learned in this unit across the curriculum, e.g. the children could use programmable toys to learn more about distance and measuring, such as how far in centimetres the toy has travelled, or to model and then solve problems in maths.

- The children could attach pencils or pens to programmable toys to make them draw shapes on paper.
- The children build on programming work in *Unit 2.1 – We are astronauts*.

 4

Assessment guidance

Use this page to assess the children’s computing knowledge and skills. You may wish to use these statements in conjunction with your own school policy for assessing work.

ALL CHILDREN SHOULD BE ABLE TO:

- Follow instructions to move around a large space
- Record a set of instructions for a toy
- Program a toy to move by giving one instruction at a time
- Program a toy to move by giving a set of instructions

MOST CHILDREN WILL BE ABLE TO:

- Give one another instructions to move around a large space
- Understand input, program and output in the context of a robotic toy
- Create a program to move a toy to a particular location
- Debug a program

SOME CHILDREN WILL BE ABLE TO:

- Predict where a set of instructions will take a pupil moving in a large space
- Predict where a toy will end up from a set of instructions
- Understand input, program and output in more general contexts
- Look for ways in which a program could be made more efficient

COMPUTING PoS REFERENCE

- Understand what algorithms are
- Create simple programs
- Programs execute by following precise and unambiguous instructions
- Create simple programs

- Understand what algorithms are
- Programs execute by following precise and unambiguous instructions
- Create simple programs
- Debug simple programs

- Use logical reasoning
- Use logical reasoning to predict the behaviour of simple programs
- Recognise common uses of information technology beyond school
- Debug simple programs

PROGRESSION

The following units will allow your children to develop their knowledge and skills further.

- *Unit 2.1 – We are astronauts*
- *Unit 2.2 – We are games testers*

Step 6: Debugging

RESOURCES

- Program that needs debugging
- Unit poster - Can you debug the program?



- Grace Hopper: http://en.wikipedia.org/wiki/Admiral_Grace_Hopper#Anecdotes



- Show pupils the map again, and the faulty set of instructions (see Resources). Explain that the algorithm is meant to move the robot from one point to another, but that you think it might be wrong. Ask pupils to predict what would happen if it was typed in as a program. Where would the robot end up? Can they suggest any way to fix the program? Explain that this is called debugging. Tell the story of Grace Hopper literally 'debugging' one of the early computers by removing an insect from a relay switch (see Resources).
- Try one or more of the pupils' suggested corrections for the algorithm, inputting this as the program for the robot. Did it run correctly? Are further changes needed?

POSSIBLE OUTCOME FOR THIS STEP:



- Groups of pupils could race one another to the treasure – from being given a start and end point, through programming their robot to running the program against the clock or head-to-head on the same or parallel maps.
- Ask pupils to discuss with their parents or carers examples of programs they use on computers or other devices that don't work exactly as they should. Are these bugs?



Let's learn

Let's try

Activity

SCHOOL

HOME

Core Steps

Step 5: Programming the robot to find the treasure

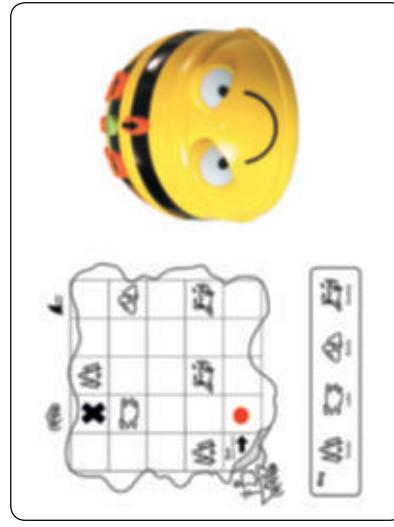
RESOURCES

- Treasure map



- <https://scratch.mit.edu/projects/121768826/>

POSSIBLE OUTCOME FOR THIS STEP:



Pupil's book

PROGRAMMING THE FLOOR ROBOT TO FIND THE TREASURE



Let's learn

Let's try

Extensions

SCHOOL

- Give each group a counter that represents the programmable toy. Ask each group to place the counter on the dock and move it one step at a time until it reaches the X, making a note of each step the counter takes (using a pencil and paper, or a laptop/PC, if appropriate).

- The groups should then see whether they can find the treasure using the programmable toy on the large treasure map. First, ask them to place the toy on the dock and the X in the square of the large map that relates to the A4-sized hard-copy version of the map. Ask them to input the sequence of instructions they have noted down and then test whether the instructions lead the toy to the treasure.

- Change the starting location and/or the location of the treasure as groups solve each problem successfully, or encourage them to set one another programming challenges, such as moving from one place to another on the map.

HOME

- The children could use the software associated with the programmable toy or the custom Scratch blocks to program it to move on-screen (see the Scratch file in Resources).

- If pupils have access to iOS tablets or phones at home, they could play with the free Bee-Bot app on these.

Step 4: Introduction to programming

RESOURCES

- Treasure map
- Blackbeard's instructions (1 and 2)

POSSIBLE OUTCOME FOR THIS STEP:

INTRODUCING THE ROBOT

Let's try



SCHOOL

- Gather the children round the large drawn-out map and explain that Captain Blackbeard buried two chests of treasure at different points on the island. Read the instructions that he wrote to remind himself where he buried the first chest of treasure (see Blackbeard's instructions 1 and 2 on setveintiuno.com). Work as a class to follow Blackbeard's instructions – with either yourself, or one or more of the children, moving 'Captain Blackbeard' (a small toy figure) around the map to discover the place where he buried his treasure.

HOME

- Ask pupils to think about whether the algorithms they've been using are the most efficient way possible to get to the destination.
- Explain that the programmable toy is going to be a treasure hunter, and if they program it with Blackbeard's instructions it will go to the treasure. Place the programmable toy on the dock and ask one or more pupils to input Blackbeard's instructions as a complete sequence in one go. Ask the children where they think the toy will end up. Show them if they are right. Ask them how they made their predictions.
- Repeat with the other set of instructions and/or a different starting point, again asking pupils to predict where the toy will end up when programmed. One or more children should program the instructions so that the class can test their predictions.
- Working in groups and using the treasure map, the children could write a sequence of instructions to a point on the map of their choosing and then other children could first predict, and then test, where the instructions take the programmable toy.
- Provide some time for pupils to experiment further with the toy, thinking of more complex programs they could give it and, in each case, making a prediction of how the robot will respond.

Core steps

Step 3: Introducing the robot

POSSIBLE OUTCOME FOR THIS STEP:



- Introduce your chosen programmable toy to the children. Provide them with some time to experiment with this, discovering how it works for themselves, particularly if it is a toy with which they are unfamiliar. Ask the pupils to demonstrate what they've discovered to the class. Correct any misconceptions. Show them what buttons to press to make the toy move one step at a time, if they have not discovered this for themselves. Allow the children time to take turns doing this (either within a group or in pairs).
- Explain that the toy is a computer – it accepts inputs and then produces output according to a stored program. Ask the children to think of other technologies that also accept input, store programs and produce outputs. Correct any misconceptions.

Pupil's book

INTRODUCING THE ROBOT



Let's learn

Extensions

SCHOOL

HOME

- The pupils could write down their instructions.

- The pupils should look for other devices around their homes that accept inputs, store programs and produce outputs, such as games consoles, smartphones, tablet computers, microwave ovens and washing machines.
- The pupils could write down their instructions.

- Show how the robot can be programmed interactively (i.e. one step at a time) to move about the island map. Ask the pupils to use step-by-step programming to move it from one place to another.

- Assign pupils to groups and ask them to set and respond to similar programming challenges themselves, moving the robot from one location to another.

- Show the pupils a set of instructions to move from one place to another, and ask them to predict where the robot will end up. Were they right? How did they work this out? Working in their groups, the pupils could set and respond to similar challenges themselves, saying a set of instructions and then all making a prediction of where the robot would end up when these instructions were programmed in, one at a time.

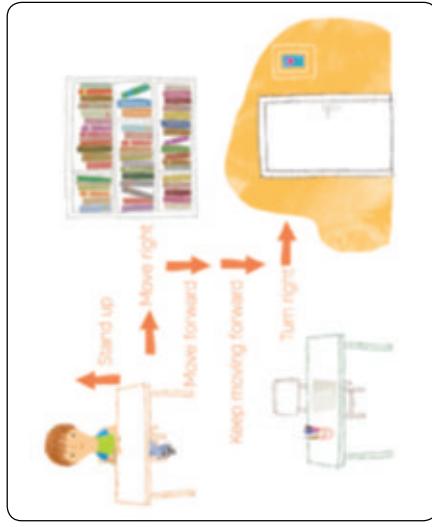
Step 2: Recording an algorithm

RESOURCES

- Treasure map
- Symbols template



POSSIBLE OUTCOME FOR THIS STEP:



- Project the treasure map PDF file (from SET VENTI UNO) onto the IWB. Discuss what the children can see. What do they think the symbols on the map represent?
- Gather the class around the large map (drawn out on paper or cardboard). Introduce a small toy or figure to the map and explain that the children need to give the figure instructions to move from one point on the map to another, e.g. from the lake to the trees. Talk about what instructions they might give to do this, e.g. forwards, backwards, turn left and turn right.
- Put a toy figure on the dock, on the large map, and tell the children where it needs to get to. Working as a class, the children should give the figure instructions to move, one step at a time, to its destination. As the figure moves, write each instruction on the IWB. The class may wish to agree a common format for this, either using symbols, words, or both. For example, to move forward one step could be recorded as: **↑ 1**, Forward 1 or FD1.
- Once the figure has reached its destination, place it back on the dock and follow the list of instructions that has just been recorded. Ensure that the children understand that using this list of instructions will always mean that, when starting from the dock and facing east, the figure ends up at the same place.
- Now ask the pupils to write down instructions to move the figure from the dock to a different location, this time writing these down all at once rather than working them out as they go. Explain that they need to think carefully ('use logical reasoning') to work out where the toy will be, and which way it will be facing after each step. Ask the children to follow the instructions that they wrote down. Did they work? If not, the pupils should correct ('debug') these and start again.



Let's try

RECORDING AN ALGORITHM

SCHOOL

- The children could explore the instructions given by Google Maps for moving between two places, such as the school and the local library: www.google.com/maps.

HOME

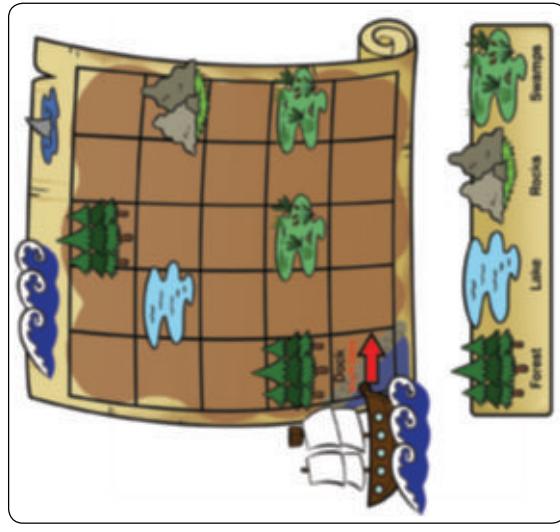
- Encourage the pupils to talk to their parents or carers about common sets of instructions they use at home, such as for getting ready for school.

3

Running the task – We are treasure hunters

Step 1: Hunting for treasure

POSSIBLE OUTCOME FOR THIS STEP:



- Take the pupils into a large space, such as the school hall or playground. Put treasure (perhaps a box or a bag) at one location and gather the pupils at another location.
- Set pupils the challenge of deciding on what instructions they'd need to follow to get from where they are to the treasure. Tell them that they can only use 'forwards', 'backwards', 'turn left' and 'turn right' as instructions. Spend some time reinforcing the ideas of left and right if necessary.

- One pupil should take on the role of the robot-pirate. Another pupil (the programmer) should read out their instructions to the robot-pirate, who should follow these exactly. Did they get the treasure? Did the programmer change their instructions at all as they saw what was happening, or did they stick exactly to their plan?
- Repeat the exercise, moving either the treasure or the pupils to a new location. This time, use an audio recorder to capture the programmer's instructions. Ask the other pupils to predict where the robot-pirate will go (they could even stand in the place they think he or she will go), and then play these back to the robot-pirate. If time permits, pupils could repeat the exercise themselves in small groups.
- Explain that the instructions are called an algorithm. When we enter these into a computer, this is programming.

Pupil's book



RECORDING AN ALGORITHM

Let's learn

Let's try

Extensions

SCHOOL

- The pupils could record their instructions in words or pictures.

HOME

- The pupils could repeat the same exercise at home, either programming, or being programmed, by their parents or carers.

→ 2 Getting ready

THINGS TO DO

- Read the Core steps sections of *Running the task*.
- Decide which programmable toy you are going to use. Programmable toys that allow children to program single instructions and a sequence of instructions for movement are suitable for use in this unit. If possible, the programmable toy should be limited to make only 90-degree turns (or multiples thereof).
- Think about the individuals and groups you have in your class. Could you use any of the *Extensions* on pages 14–19 to extend your more able children? Could you use any of the suggestions in *Inclusion* (see below) to support children with specific needs? Have you considered how a Teaching Assistant will support you and the children, if one is available?
- Make the necessary arrangements for work on the field, playground or in the hall.
- Check that batteries for your programmable toys are charged, and that you have sufficient spare

batteries. Where batteries are used, the battery cover should always be screwed in place.

THINGS YOU NEED

- Printed copies of the treasure map (see setveintiuno.com)
- The treasure map drawn out on paper or cardboard to mirror the treasure map on SET VEINTIUNO. The size of the map grid should equate to the distance your programmable toy moves in one step. For example, the Bee-Bot and Roamer Too move in 15 cm steps, so the corresponding map grid would measure 75 x 75 cm. A sheet of large landmarks and an X is provided on the web. The landmarks can be pasted to the map and the X can be mounted on cardboard
- Pencils
- Programmable toys and any associated software
- Computers/laptops/tablets, with appropriate software installed for *Extensions*
- Counters (see Step 3 of *Running the task*, page 16)
- An audio recorder (or the voice memo app on your mobile phone)



RESOURCES

- Treasure map
- Treasure island Scratch file
- Symbols template
- Blackbeard's instructions (1 and 2)
- Program that needs debugging
- Unit poster – Can you debug the program?



E-SAFETY

- If children will be accessing the internet, ensure that access to inappropriate material is blocked by filters.
- If children are filmed, or film one another, working with the robots, ensure any necessary permission has been obtained.



INCLUSION

- Some children, such as those with special education needs, may need support in recording their instructions and may also need to program their toy with shorter and/or less complicated sequences of instructions. If this is the case, consider the location of the treasure in Step 3 of *Running the task*, page 16).
- Some children might have difficulty using the buttons to program the robot. Explore alternative hardware or consider how pupils may be paired/grouped to enable all to participate.



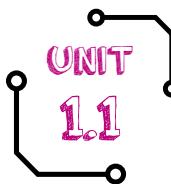
USEFUL LINKS

Software and tools

- Bee-Bots: www.tts-group.co.uk/shops/tts/Products/PD1723538/Bee-Bot-Floor-Robot.
- Roamer Too: www.valiant-technology.com/uk/pages/roamertoohome.php?cat=8&8. The Early Years keypad would suffice for this unit: www.valiant-technology.com/us/pages/foundation.php.
- The Bee-Bot app (<https://itunes.apple.com/gb/app/bbee-bot/id500131639>) is free. Although more advanced, the Daisy the Dinosaur app provides a good platform for introductory programming: <https://itunes.apple.com/gb/app/daisy-the-dinosaur/id490514278>.
- View on-screen simulators for Bee-Bots (www.tts-group.co.uk/shops/tts/Products/PD1722395/Focus-on-Bee-Bot-Lesson-Activities-1/?rguid=1c6eb9f8-7fe8-4b52-b244-891972ce6b40) and the original Roamer (www.valiant-technology.com/uk/pages/roamer_rworld.php?cat=1id0).
- The *Make a block* function in Scratch 2.0 allows you to create your own simulator of a floor turtle. See <https://scratch.mit.edu/projects/121768826/>.

Online tutorials

- Bee-Bot: www.youtube.com/watch?v=52ZuenJFyE.
- Roamer Too: <http://vimeo.com/49152214>.



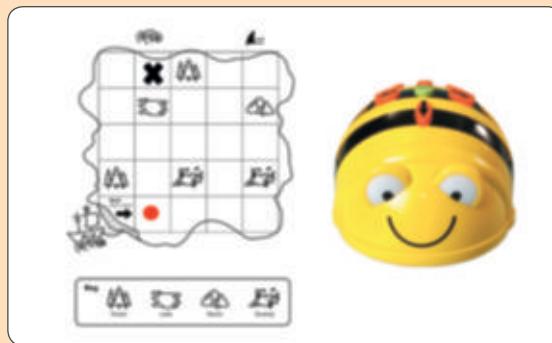
We are treasure hunters

Using programmable toys

1

About this unit

- Software:** Programming interface for programmable toy
Apps: Bee-Bot app; Daisy the Dinosaur
Hardware: Programmable toy, such as a Bee-Bot or Roamer
Too. Audio recorders are needed for the first step (your phone may be sufficient)
Outcome: A sequence of instructions that will move a programmable toy along a given route



UNIT SUMMARY

In this unit, the children will program a toy to move around a map to find buried treasure. They will start by thinking of algorithms for their routes, then input these as stored programs for the robot. They predict how the robot will move and will debug their programs.

CURRICULUM LINKS

Computing PoS

- Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions.
- Create and debug simple programs.
- Use logical reasoning to predict the behaviour of simple programs.
- Recognise common uses of information technology beyond school.

Suggested subject links

- **Geography:** Children use geographical language, and could use and make their own maps.
- **Maths:** Children recognise movements in a straight line (translations) and rotations, and combine them in simple ways. They start to recognise and make whole, half and quarter turns, and learn to recognise a right angle.
- **History:** Children find out about the past.
- **Social science:** Children could consider the effect that pirates' actions had on others.
- **English:** Children provide clear instructions for moving around a map.

TRANSLATING THE COMPUTING PoS

- In creating their *instructions*, pupils are creating *algorithms* – step-by-step instructions to achieve a particular goal. *Programming* involves converting these instructions into a formal language

understood by the computer – on this occasion, this is a series of button presses on a toy robot. The robot simply follows the *instructions* it's given.

- Sometimes, pupils' *programs* won't work as planned. In these cases, they need to correct (*debug*) their programs to fix their mistakes. When looking at others' algorithms or programs, pupils should have a clear idea of what the program will do by using *logical reasoning* to predict what will happen from the instructions.

LEARNING EXPECTATIONS

This unit will enable the children to:

- understand that a programmable toy can be controlled by inputting a sequence of instructions
- develop and record sequences of instructions as an algorithm
- program the toy to follow their algorithm
- debug their programs
- predict how their programs will work.

The assessment guidance on page 20 will help you to decide whether the children have met these expectations.

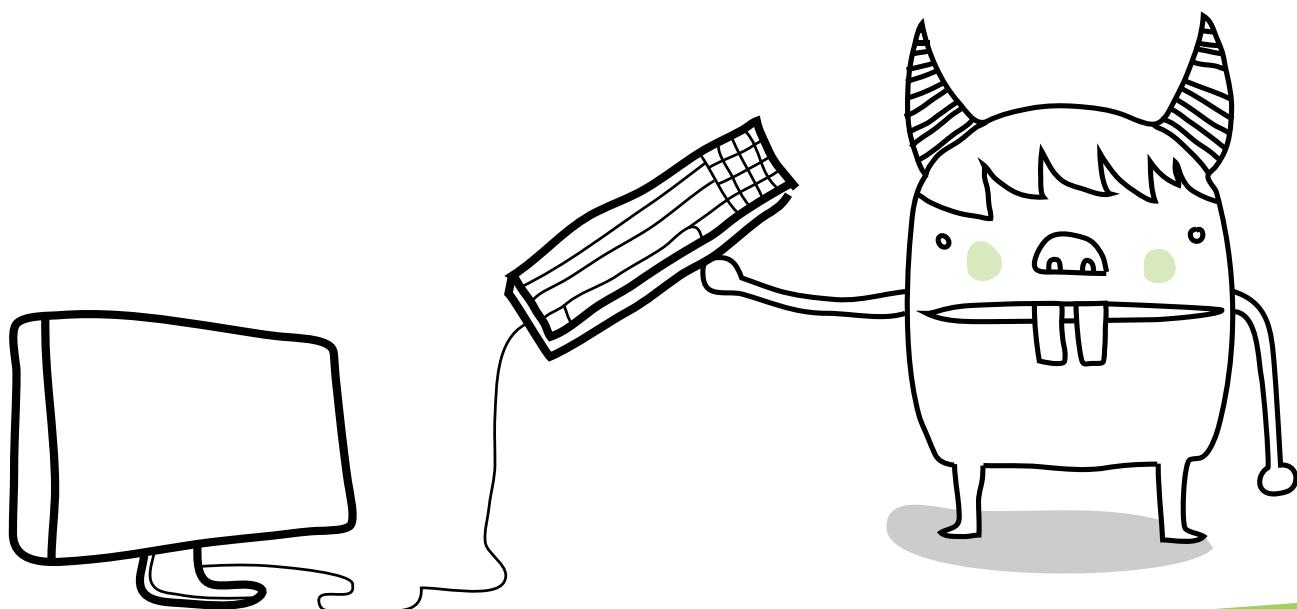
VARIATIONS TO TRY

- Children could instruct a programmable toy to conduct a tour around a plan of the school.
- Children could make the robot a delivery person and instruct it to move between photographs or models of landmarks from their local environment.
- Linking with English, children could move the toy robot around a letter grid to spell simple words.
- Children could use the programmable toy for counting, and addition and subtraction exercises.
- Many programmable toys have on-screen equivalents, or you can make your own simulator using Scratch 2.0.

**SET
VEINTI
UNO**



SWITCHED ON COMPUTING



Year 1
6-7 year olds
Teacher's book

My progress



Name:



Class:

Insert a picture of
your progress here!

1.1 We are treasure hunters

I can follow instructions.



I can record a set of instructions.



I can program a toy.



I can give instructions.



I know what input, program and output means for a robot toy.



I can give examples of input, program and output.



I can create a program.



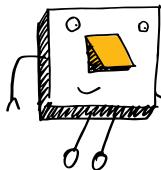
I can spot and correct mistakes in a program (debug).



I can predict where a set of instructions will take a toy or person.



I can look for ways to make a program work better.



Let's reflect

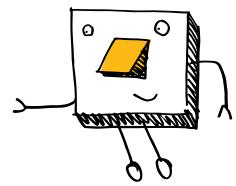
I am good at ...

Next time I will ...

Let's try



Try inputting the corrected algorithm into a floor robot. Do you need any more changes? Discuss with your partner.



Activity

Program two floor robots to dance to a piece of music.

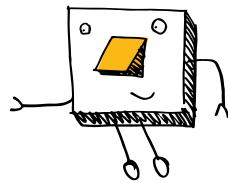
Think carefully about the parts where you want the floor robots to be moving together. Also, think about the parts where you want their movements to be a little different, such as turning in different directions.



Let's try



Read through the wrong instructions below with a partner.



Instructions:

Is this how to get to the forest?

1. R1
2. FD2
3. L1
4. FD2

Is this how to get to the rocks?

1. R3
2. FD2
3. R1



Predict what will happen if these instructions are programmed into a floor robot? Discuss it with a partner.



Suggest ways to fix the program to your partner. Make a list below.

4

Debugging

Let's learn

Debugging is a really important part of programming. Sometimes programs do not run well because there are mistakes (or 'bugs') in our programming. Debugging means checking programming and correcting mistakes so it does work.

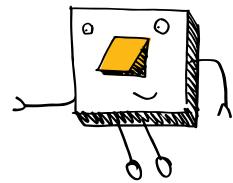
One useful way to debug programs, is to read through the instructions and decide what you think will happen. Making a **prediction** can help you to see errors. Getting a partner to look over your instructions and explaining it to them can help to find mistakes, too.



Let's try



- Put a counter at the start of the treasure map. This represents the floor robot. Move it one step at a time until it reaches the **X**.



- Make a note of each step the counter takes below. Use the instructions FD, BD, R and L.



- Put a floor robot on the starting point of a bigger version of the map. Program the instructions you wrote above into the floor robot in one go. When the program is run, does the floor robot move to the **X** on the map? If not, what part of your program should you change?



- Change the starting location of the floor robot. Program it to move to the **X** again.



Programming the floor robot to find the treasure

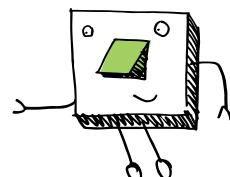
Let's learn

Look at the treasure map below. Can you spot the **X** symbol?
This represents more buried treasure.



When you wrote instructions for moving around a map before, you used arrow symbols. You can write instructions for moving around a map in different ways. For example:

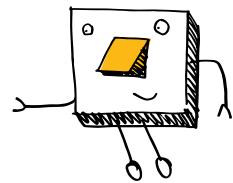
- ➊ *FD1* could represent *one step forward*. The number could be changed to show a different number of steps forward, e.g. *FD4* would mean *four steps forward*.
- ➋ *BD1* could be *one step backward*.
- ➌ *R* and *L* could mean *turn right* and *turn left*.



Let's try



Work in a group to follow Captain Blackbeard's instructions on the map using a counter.



Record Captain Blackbeard's instructions below using the arrow symbols.

- ↑ 1 = one step forward
- ↓ 1 = one step backward
- = turn right
- ← = turn left

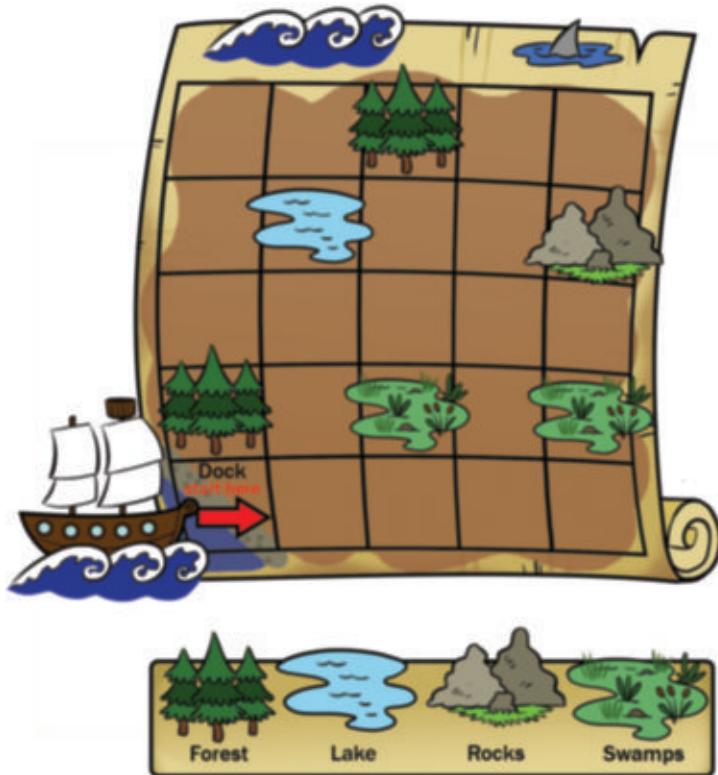
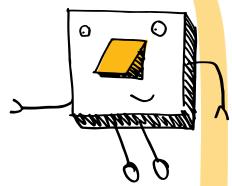


Write a sequence of instructions, using either words or symbols, to a secret place on the map. Get a partner to follow your instructions with a counter. Did they arrive at the correct place?

Let's try



Look at the treasure map below.



The pirate Captain Blackbeard has buried two treasure chests in different places on the island. He wrote these instructions to remind himself where he buried the first chest. Read through them.

1. Take 1 step forward.
2. Turn left. You will see the lake in the distance.
3. Take 2 steps forward.
4. Turn right.
5. Take 2 steps forward.
6. Turn left.
7. Take 1 final step.
8. Start digging!

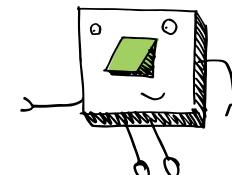
2

Introducing the robot

Let's learn

Toys that can be programmed, like floor robots, are actually **computers**. This is because they accept **input** and produce an **output** according to a program. They can be programmed to move from one place to another. You can try to follow **instructions** to move around a map like a floor robot.

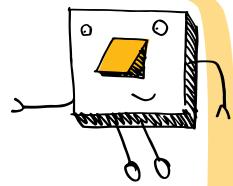
Other technologies also accept inputs, store programs and produce outputs, such as smartphones, tablets and even household devices like this microwave.



Let's try



Write an algorithm that would direct a robot/pirate from the lake to the forest. Use the arrow symbols below to show your algorithm.



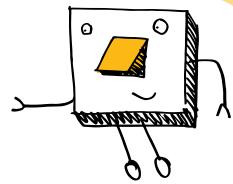
- ↑ 1 = one step forward
- ↓ 1 = one step backward
- = turn right
- ← = turn left



Ask your partner to follow your algorithm. Does it instruct them from the lake to the forest? If not, you will need to debug (change) your algorithm and make sure it works.



Let's try



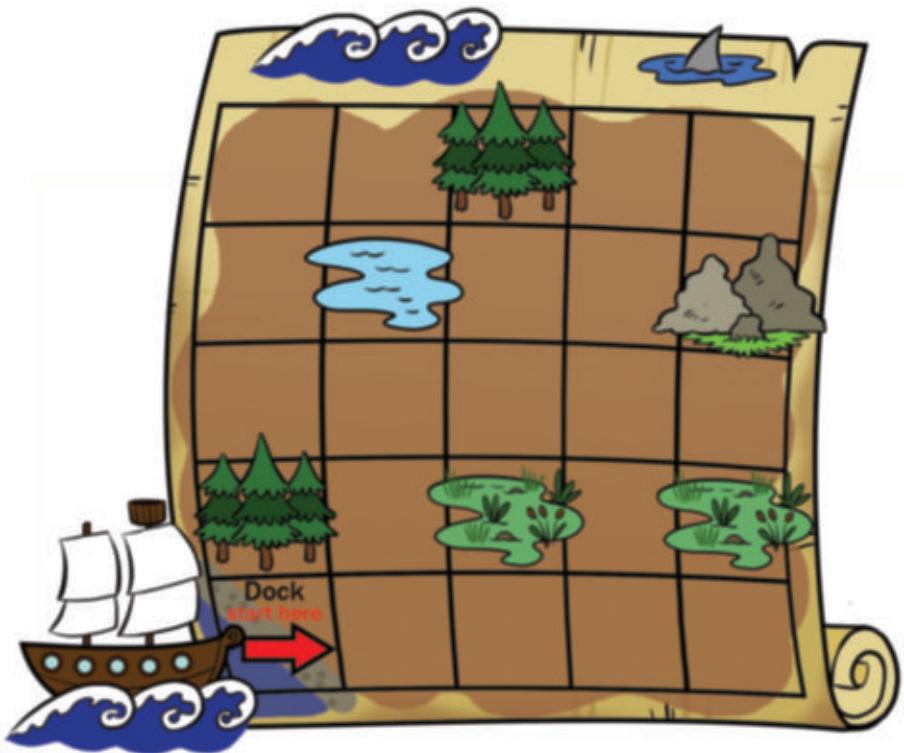
Work in pairs to create algorithms that tell people to move from one part of the room to another.

One of you needs to be the robot (the person listening and following the instructions). The other person needs to be the programmer (the person giving the instructions). Take turns.

You can only use 'forwards', 'backwards', 'turn left' and 'turn right' as instructions.



Look at the treasure map below. Discuss with a partner what you think the symbols mean.

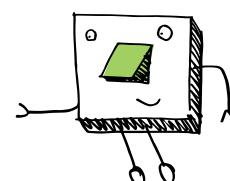
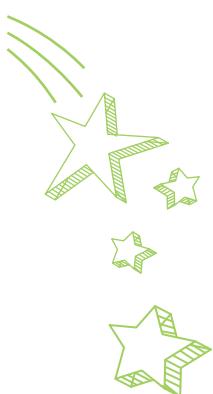


Algorithms are instructions that can be followed to solve a problem or complete a task. When algorithms are entered into a computer or a device like a floor robot, it is called **programming**.

Algorithms are written in **languages** that people can understand. Programs are written in languages that computers can understand. These languages are often known as **code**.



Image of computer code.

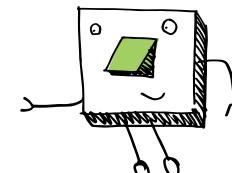
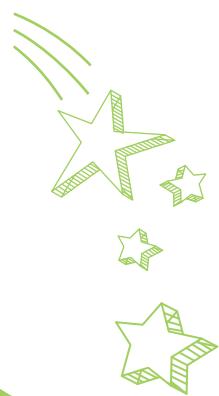
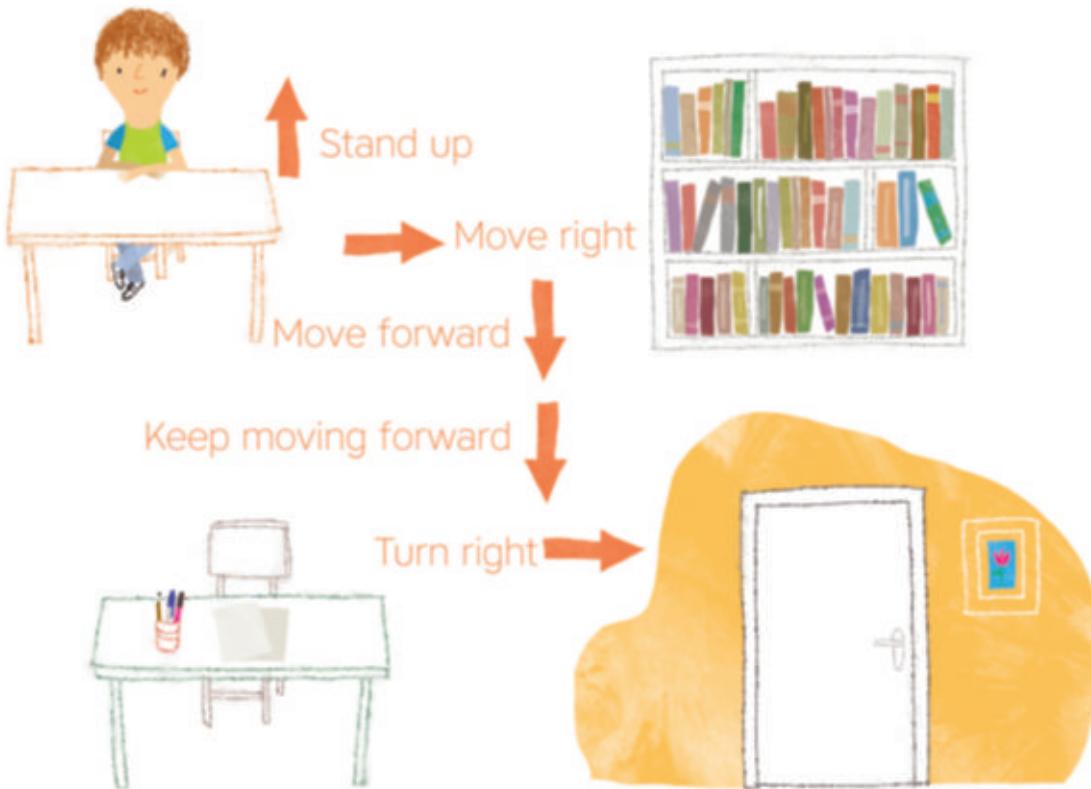


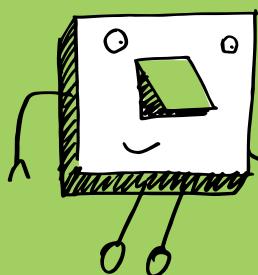


Recording an algorithm

Let's learn

What instructions will you need to follow to get from where you are sitting to the door of your classroom?





1.1

We are treasure hunters



In this unit you will:

- ⌚ Program a toy to move around a map to find a buried treasure
- ⌚ Learn about algorithms
- ⌚ Predict how your robot will move
- ⌚ Debug your programs



Key words

Algorithms: a set of instructions to solve a problem or complete a task

Programming: changing the instructions in your algorithm into a language your robot can understand

Predict: saying what you think will happen

Data: facts and other information

Input: data that is sent to a computer

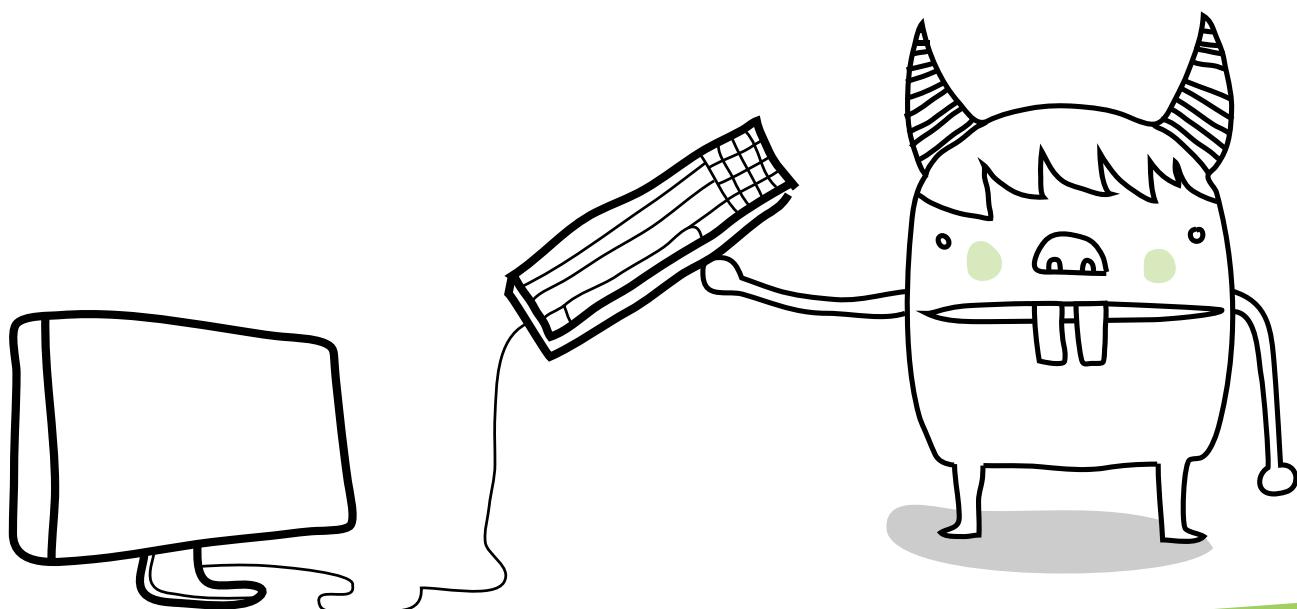
Output: data that is produced by a computer

Debugging: finding and correcting mistakes in your programs

**SET
VEINTI
UNO**



SWITCHED ON COMPUTING



Year 1
6-7 year olds
Pupil's book

Demo project: We are programmers

In this unit, the children create an animated cartoon using characters they design. They use a paint tool to create characters and backgrounds.

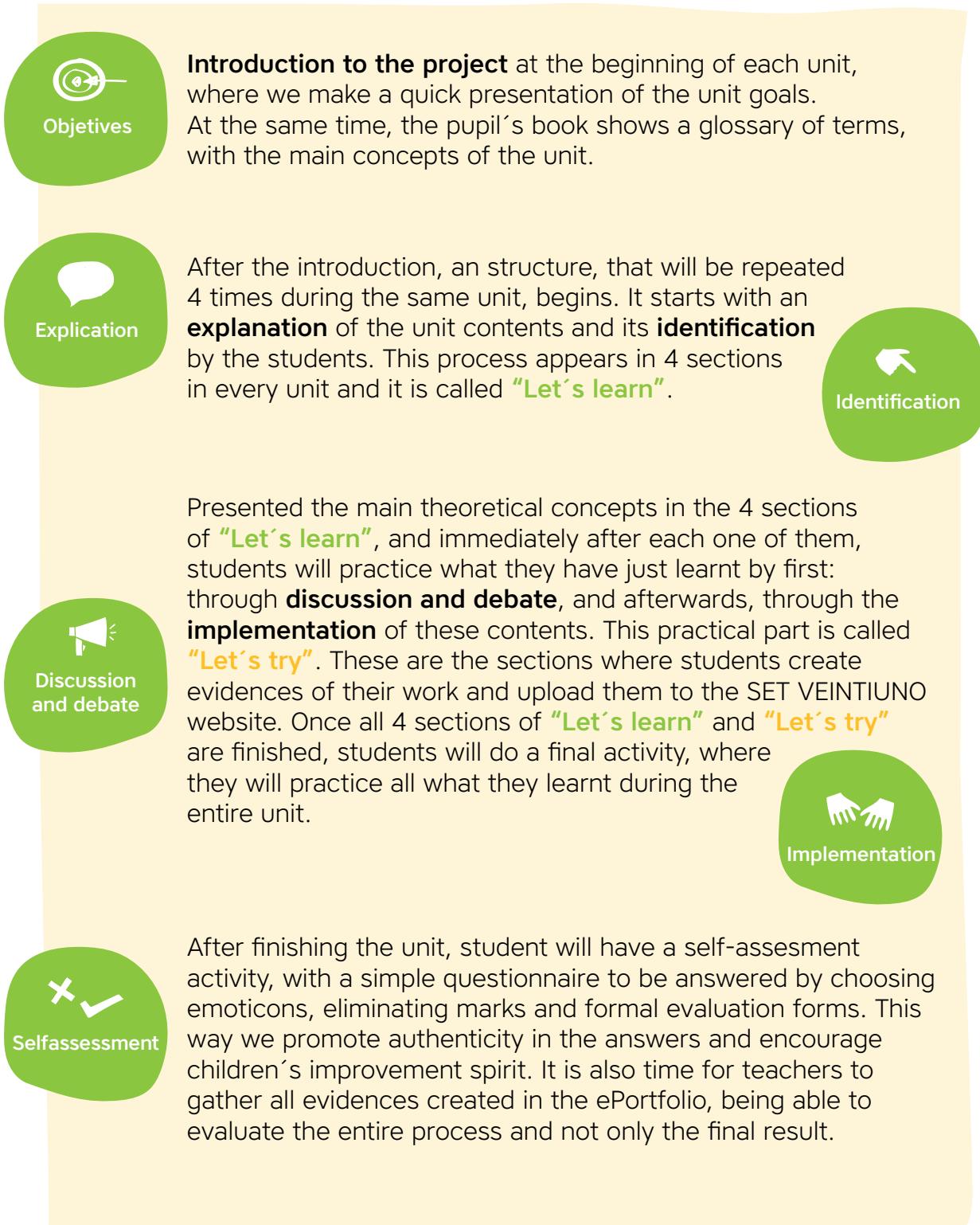
They then create an animation by translating a storyboard into a series of scripted instructions (program) for graphic objects.

Students use a drafting tool to create their characters (if they haven't done it yet) and the backgrounds. Then, they will program an animation transforming a storyboard in a series of instructions for graphic objects. Students will this way work variables and divers methods of inputs and outputs, as well as train their logical reasoning to debug algorithms and programmes, an absolutely indispensable skill to develop computational thinking. In addition, they will be able to choose an specific software for a specific duty and combine it with another software to create their animation.

It is an ideal moment to present their work in front of the class and practice their public speaking.



SET VEINTIUNO's *Switched on Computing* repeats the content structure in every level, adapting each of them to the features of the different aged students, in order to consolidate an effective and practical process:



Keys to understand Switched on Computing

1

Introduction to the **STEAM** world. Handling of the main digital tools, first steps in coding, understanding of the functioning of the most common digital devices and applied technology creation, involving arts and creativity in the process.

2

Safe, responsible and ethic use of ICT. Promoting empathy and citizen values. Online-safety advices and highlighting subjects such as copyrights, licences and permissions needed to publish class-work.

3

Computational thinking related skills development: logic, abstraction, breaking down of problems, algorithm creation or debugging, among many others.

4

Learning by doing inside a project-based learning system. Encouragement of **experimentation as the motivating element**. All units are structured on a way that we first explain the theory and immediately after, we practice it, taking a step forward to the accomplishment of the challenge the unit presents.

5

Development of **perseverance, resilience and tolerance to the frustration** in the search of a specific goal. We work with: plenty of debugging activities, normalizing mistakes and their correction; and activities involving criticizing of each others work and the own.

6

Promotion of **creativity**, leaving room to **imagination** with plenty of statements like: "Now create an animation of a joke using Scratch".

7

Elimination of cultural gender roles associated to the ICT world through the introduction of STEAM project from an early age. Those children that start creating technology since Primary school, will see with their own eyes there are no differences regarding the performance of boys and girls, disposing the idea of technology being a thing for men.

8

Constant promotion of **collaborative learning**, both through methodology and the use of open software, from where the students will get other's people materials (always meeting the copyrights and necessary permissions) and share their work with the online community.

Shift in the evaluation model:

- Promotion of the **evidence-based evaluation**.
- Works with **self-assessment and co-evaluation** in between students.

9

Rise of the students' **self-esteem**, as they become technology makers apart from mere users. Students can see how they are able to create fully functional software, useful to them and their communities, empowering them to apply their new skills to any project they want. The goal is to make them see that, if they want it, they can do it.

Switched on Computing SET VEINTIUNO

We are immersed in an exciting new era of change. A new era where technology and robotics are becoming an absolutely indispensable part of our daily life. A daily life that brings with it an enormous amount of challenges but also opportunities. It is in our hands to prepare our youth for these opportunities.

For this purpose, Santillana created *Switched on Computing*, a computational thinking and digital literacy programme for Primary Education, promoting an ethic and responsible use of technology.

Switched on Computing is programme that has been running already for three years in the international market, showing excellent results, which Santillana licensed and adapted for us. It introduces computing science into the context of the student, making it attractive and closer, as well as practical for them, due to the suitability of the contents to the students' daily life, from the very first minute. The programme includes a very detailed teacher's book that allows any teacher to follow, step by step, all units and levels of *Switched on Computing*, no matter what previous knowledge they might have.

By introducing computing science at such an early age (from 6 years old), we contribute not only to instil an ethic and respectful use of technology, but also to eliminate all gender stereotypes related to ICT. By working computing science with the youngest, we erase those preconceived ideas of technology being for boys only and make our part on matching the percentage of women in technical jobs, which is currently a very shy 20%.

Switched on Computing has 6 levels, one per each year of the primary school, with 6 units consisting of 6 sessions of 45 minutes each. Its materials are available both in English and Spanish and are adapted to the features of the different ages between the Primary education.

Apart from the students' and teacher's books, these materials include a series of videos, templates and many other online items inside our website SET VEINTIUNO. com, which is not only a repository of extra contents, it is an essential part of the programme as it allows teachers to collect evidences of their students' work. This evidences need to be uploaded to our web's ePortfolio, which allows to evaluate not only the final results, but also the entire process on a simple click. The opportunity to share the work and ideas with other members of the SET VEINTIUNO community is also provided.