

8. Scratch: creación de animaciones con niños para desarrollar pensamiento estructurado*



Scratch es un programa que permite trabajar con niños de manera simple, pero eficiente, animaciones sencillas donde se integran escenarios, personajes y elementos sonoros, sea creados por el usuario en el entorno de trabajo, elaborados en otros programas, o tomados de la galería de recursos. Todo esto para presentar una opción accesible y atractiva para iniciar a los niños en la lógica de un lenguaje de programación básico, desarrollando su creatividad, capacidad de análisis, interpretación, síntesis y argumentación, con lo que se estimula y pone en marcha un pensamiento lógico y estructurado. En consecuencia es una excelente herramienta para la escuela, pudiendo a través de ella desarrollar procesos transversales de aprendizaje.

El programa, de uso gratuito, fue desarrollado en 2007 por un grupo de investigadores del Lifelong Kindergarten Group del Laboratorio de Medios del MIT (Instituto de Tecnología de Massachusetts), dedicado a la convergencia de la multimedia y la tecnología. El apoyo financiero fue recibido de parte de la National Science Foundation, Microsoft, la Fundación Intel, Nokia, lomega y el consorcio de investigación del Laboratorio de Medios del MIT.

A través del sitio web oficial se accede a ejercicios realizados en Scratch <http://scratch.mit.edu/>.

8.1. *Qué caracteriza este programa*

Scratch permite que los niños puedan generar una sucesión de acciones, encajando bloques gráficos, los cuales se pueden definir como una orden que se le da a un objeto o imagen, por ejemplo “Mover 10 pasos”. Los bloques se encajan como piezas en un juego de armotodo, construyendo la sintaxis adecuada que da lugar a la animación.

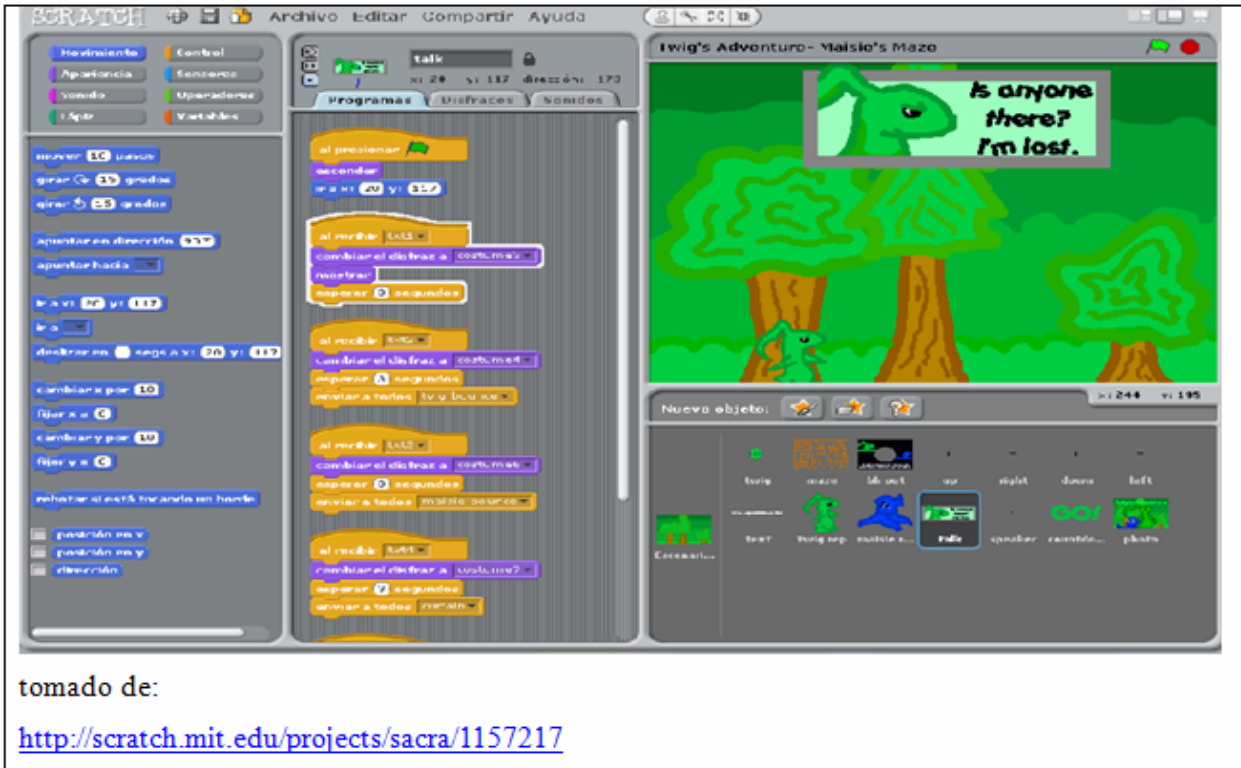
Entre las características llamativas de Scratch, a parte de la manipulación de imagen, sonido y video para construir las animaciones (secuencia de procedimientos o acciones), está su posibilidad de compartir proyectos a través de la comunidad de Scratch en Internet, pudiendo intercambiar no sólo los productos finales sino también gráficos e insumos en general. De esta manera fomenta la participación en una comunidad de colaboradores en torno a las ideas que se trabajen. Este aspecto se ve reforzado en la característica adicional de poder cambiar el idioma de los bloques, incluso durante la ejecución de un proyecto, sin que esto represente algún problema.

Se puede resumir entonces del diseño de Scratch que:

- Utiliza la metáfora de los bloques de construcción para desarrollar secuencias de acciones.

* Por: Ennio Erazo, Ingeniero de Sistemas y Técnico en Gestión Empresarial

- Permite la manipulación y construcción de animaciones con recursos multimedia: sonido video e imágenes.
- Facilita la portabilidad e intercambio, ayudado por la facilidad de contar con diferentes entornos lingüísticos.



Adicionalmente, se consideran entre sus ventajas:

- Es gratuito, permitiendo a las instituciones integrarlo sin limitarse por los costos de licencia.
- Facilita al docente desarrollar en los estudiantes un pensamiento creativo, crítico y reflexivo, a través de una herramienta atractiva y motivante.
- Es susceptible de ser usado en todas las áreas de conocimiento, incluso integrándolas, de manera que posibilita una enseñanza transversal.
- Propicia abordar temáticas de forma creativa posibilitando que el estudiante se entusiasme e involucre en desafíos y oportunidades de aprender, incluso en tareas intelectuales exigentes.
- Es multiplataforma, lo que significa que hay versiones para Windows, Mac y Linux.
- Permite compartir los proyectos en línea, propiciando un ambiente colaborativo.
- Los materiales elaborados se pueden insertar en una página web.
- Tanto el entorno de desarrollo como las ayudas se encuentran disponibles en diferentes idiomas, incluido el español.

Entre las desventajas podría mencionarse que los archivos generados, especialmente aquellos que involucran muchos elementos, quedan muy pesados, lo que hace que a veces el programa no

responda con la rapidez deseada. Esto depende también de las características del equipo en que se esté corriendo y, al verlo por Internet, de la velocidad de conexión.

8.2. Qué procesos de aprendizaje impulsar con Scratch

Los conceptos de enseñanza y aprendizaje han cambiado sustancialmente durante los últimos años. Hoy día es común hablar del aprendizaje significativo, enmarcando con ello una dimensión de pensamiento y creación intelectual que busca que el estudiante no sólo interprete y analice una temática sino que además sea propositivo en torno a ella, trascendiendo un esquema simplemente memorístico. Es así que los educadores están frente al reto de actualizarse permanentemente sobre metodologías y recursos que apunten en esta dirección, adoptándolas y adaptándolas a su contexto particular. Una de estas posibilidades, entre muchas otras, es Scratch, pero concretamente ¿Cómo incentiva Scratch en los estudiantes la interpretación y solución de problemas matemáticos, creación de textos narrativos, historias interactivas, animaciones, juegos, música, arte?

Scratch es una herramienta muy útil para el aprendizaje significativo desde diferentes áreas porque apunta a desarrollar pensamiento estructurado, lo que implica ganar aptitudes de análisis, proposición y argumentación. Esto lo logra haciendo que usuarios sin experiencia en programación se inicien, casi sin notarlo y de forma amena y divertida, en la organización de procesos de manera secuencial y lógica. En este sentido los docentes pueden abordar la identificación y resolución de situaciones con el estudiante desde un entorno gráfico llamativo, el cual sirve como medio para expresar y desarrollar el proceso. De esta manera facilita ganar habilidades lógico-matemáticas, de lenguaje y comunicativas, aunque no lo delimita necesariamente a áreas particulares.

En relación con la apropiación de conceptos matemáticos e informáticos, se puede destacar que Scratch, al igual que todas las herramientas de programación, posee funciones matemáticas definidas, incluyendo además opciones de trabajo gráfico de la mismas. En este sentido puede ser utilizado para diagramar problemas matemáticos, físicos o trigonométricos, y la solución de los mismos, con lo que se llevaría al estudiante a realizar pasos o secuencias organizadas para representar gráficamente conceptos, facilitando su mecanización, comprensión y manejo. Scratch es así una alternativa valiosa para propiciar estos aprendizajes, a la vez que se desarrolla la imaginación y creatividad, mediante:

- Aplicación de coordenadas en un plano.
- Planteamiento, gráfica, análisis y resolución de problemas de una o más variables.
- Manejo de criterios condicionales.
- Procesos interactivos (bucles).
- Control de flujo, iteración, eventos.
- Mayor competencia y apropiación de las TIC.
- Procesos de diseño (prototipo, prueba, rediseño).
- Expresión creativa, lógica, iconográfica.
- Iniciarse en la lógica de la programación.

El desarrollo del pensamiento lógico-matemático es una herramienta cognitiva fundamental para lograr un cambio en la forma de aprendizaje de los estudiantes, pues explota al máximo su

potencial creativo y propositivo al encaminar las ideas en procesos estructurados que posibilitan la acción. Puede definirse entonces como la capacidad de un individuo para generar relaciones coordinadas, lo que implica desarrollar habilidades tales como: Analizar, sintetizar, comparar, abstraer, caracterizar, definir, identificar, clasificar, ordenar, relacionar, razonar, interpretar, argumentar. El pensamiento lógico así entendido es transversal a todas las áreas, beneficiando ampliamente al estudiante en su proceso de aprendizaje.

Scratch también puede ser usado en áreas como español, ciencias sociales, ética y valores, inglés, para motivar la construcción de historias cortas animadas a partir de temas o situaciones construidas en el aula, incentivando el desarrollo de competencias comunicativas y de lenguaje. Llevar al estudiante a construir una narración, graficarla, sonorizarla, animarla, es incitarlos a ser propositivos, crear soluciones, representar sus realidades, con lo que se puede enfatizar en el reconocimiento del contexto al que pertenece, incluso en relación con otros contextos. Scratch se convierte así en una herramienta de expresión que permite indagar sobre sus aprendizajes, además de reforzarlos.

Finalmente, la construcción de animaciones integra inherentemente el desarrollo de una estética y un ritmo audiovisual, con lo que se ve involucrada la artística. La creación y exploración de escenarios, personajes y sonidos se convierte en un camino para que el estudiante participe en procesos contemplativos, imaginativos y selectivos, exprese aspectos sociales de su entorno, recree y comparta, además de estimular y encausar su sensibilidad aprendiendo a pensar creativamente en relación con situaciones de su contexto. Aspectos todos involucrados con el desarrollo de competencias desde el área.



Dialogo 1



Dialogo 2



Dialogo 3

Partiendo de lo planteado, la riqueza de usar Scratch estará en la orientación pedagógica que se construya para generar estrategias que posibiliten el surgimiento y fortaleciendo de pensamiento lógico de forma individual y grupal, contribuyendo a un ambiente donde se solucionen situaciones creativamente. El uso de Scratch como simple entorno para entretenerse, sin una intención pedagógica y comunicativa, no significará alcanzar los resultados esperados.

A continuación se hace una introducción a Scratch. Se plantean sólo algunos ejemplos, pues está en manos del docente lograr alternativas de acción acordes al contexto e intenciones educativas a partir de una exploración más a fondo.

8.3. Consideraciones para descargar e instalar Scratch

Antes de hacer la instalación es importante considerar que se requiere como requisitos de sistema:

- Sistema Operativo (S.O.): Windows 2000, Windows XP, Windows Vista, GNU/Linux, Mac OS X 10.4 o más reciente.

- Instalación previa de Java versión 6.1 o superior.
- Pantalla: 800x480 o más grande y (16-bits de color o más.
- Como mínimo 120 megabytes de espacio libre en el Disco Duro para instalar Scratch
- Sonido (opcional): Parlantes o audífonos; micrófono para grabación

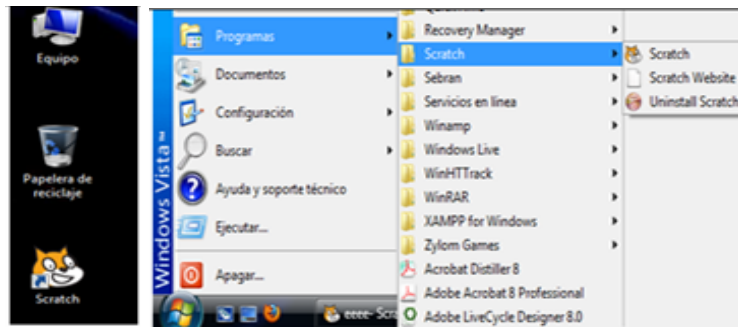
Para descargar los archivos de instalación ingrese al sitio oficial del proyecto en Internet:

<http://scratch.mit.edu/>

En este sitio se consiguen igualmente tutoriales muy completos que ayudan a reconocer a profundidad las opciones del programa.

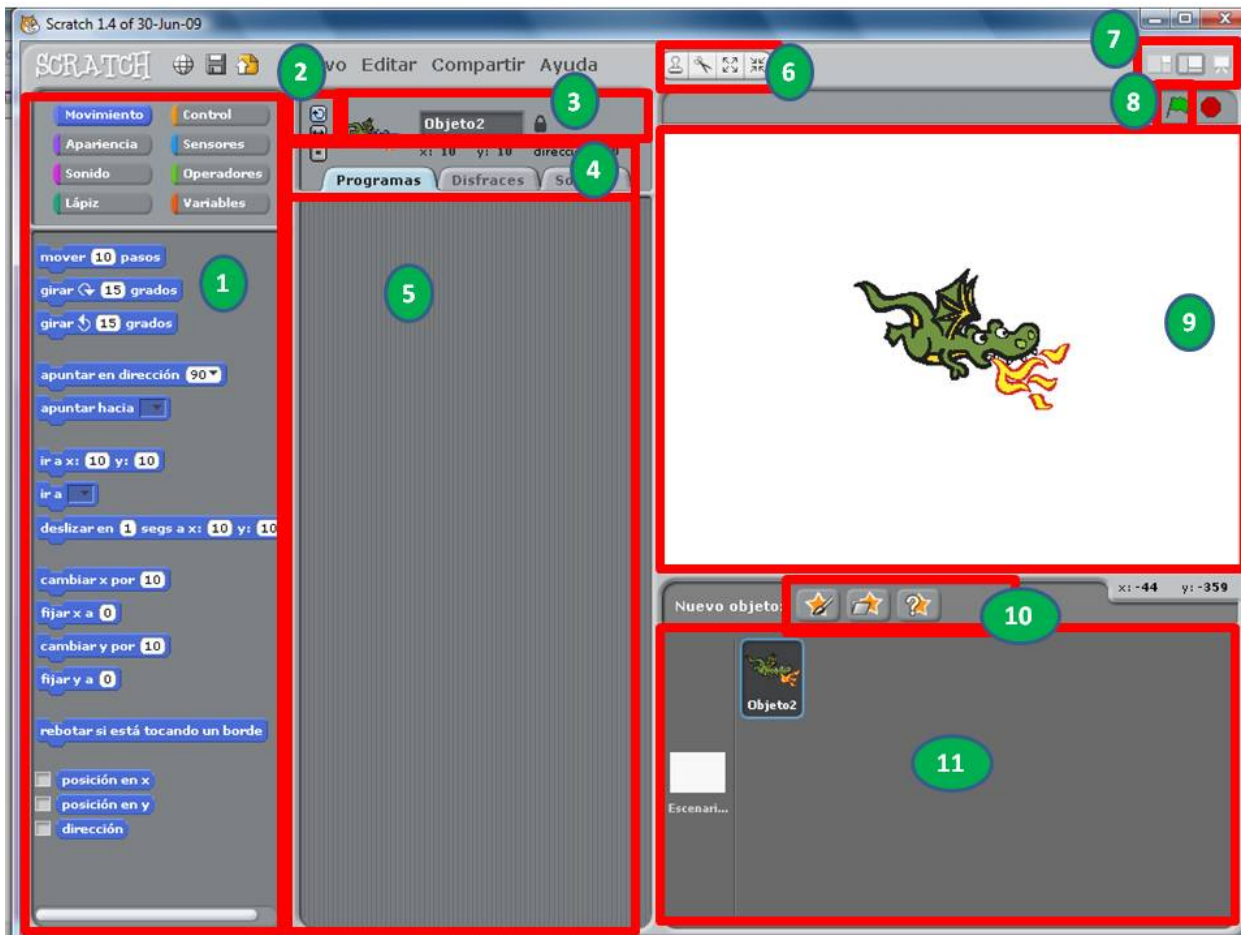
8.4. Reconocimiento del entorno de trabajo

Una vez se ha instalado el programa, se va a acceder a través del ícono que queda en el escritorio del computador, o siguiendo la ruta: Inicio/programas/scratch/scratch



Al instalar el programa en el computador tendrá acceso al siguiente entorno de trabajo. Se recomienda familiarizarse con cada una de las opciones, las cuales se explican a continuación:

1. Paleta de Bloques: Se cuenta con ocho grupos, cada uno de un color, que contienen bloques gráficos de programación representando diferentes acciones. Los grupos son: Movimiento, apariencia, sonido, lápiz, control, sensores, números y variables. Se recomienda explorar cada uno para reconocer las diferentes acciones que agrupa y son posibles con el programa.
2. Estilo de rotación: Permite determinar la rotación de los actores, los cuales se denominan sprites.
3. Información del sprite seleccionado: Muestra el nombre del actor seleccionado y una miniatura del mismo.
4. Pestañas: Despliega según se seleccione los programas, disfraces o sonidos para trabajar en ellos o editarlos.
5. Área de Scripts: A este espacio se arrastran los bloques de programación para construir los programas o guiones que dan vida a la animación. Se le llama así porque al conjunto de acciones, expresada en bloques, se le llama crear scripts.
6. Barra de Herramientas: Tiene opciones para mover los objetos, copiar, cortar, aumentar su tamaño o disminuir su tamaño.
7. Modo Presentación: Permite elegir la forma en que se quiere visualizar la animación, incluyendo ampliar el escenario a pantalla completa.



8. Bandera verde-roja: Sirve para ejecutar o detener los programas o animaciones creadas. Da, en otras palabras, las opciones de reproducción.
9. Escenario: En él se visualiza el programa. Funciona como un plano cartesiano sobre el cual se ubican los objetos indicando su posición en X y Y.
10. Botones Nuevos Objetos: Justo debajo del escenario hay tres botones que nos van a permitir buscar o incluso crear nuevos actores o elementos para nuestra acción.
11. Zona de sprites y escenarios: Muestra en miniatura el listado de los sprites del programa, permitiendo editarlos al hacer clic encima. Igualmente muestra el escenario en el que se está y permite cambiarlo. Para hacerlo, clic encima, con lo que se habilita en la zona de pestañas la opción Fondo.

8.5. Jugando y aprendiendo con Scratch en la escuela: un primer ejercicio práctico

¡Iniciemos!

Vamos a simular el movimiento de una persona, haciendola recorrer una distancia de 50 metros en 20 segundos, con el fin de encontrar su velocidad de desplazamiento. Para ello se va a tener en cuenta que la velocidad se conoce dividiendo la distancia que debe recorrer entre el tiempo que le toma hacerlo ($\text{Velocidad} = \text{Distancia} / \text{Tiempo}$). El estudiante debe preguntarse si sus conocimientos actuales le permiten resolver este problema.

Antes de plantar cómo se haría en Scratch, se va a simular esta situación desde la programación algorítmica utilizando pseudocódigo. Esto para evidenciar en la práctica a qué se hace referencia

cuando se habla de pensamiento estructurado y lógico, organización por pasos, etc. Tener claro estos conceptos ayuda a aprovechar realmente Scratch como proceso de aprendizaje, y no meramente como un entorno para jugar a generar pequeñas piezas con movimiento.

El pseudocódigo es un lenguaje de programación algorítmico, intermedio entre el lenguaje natural y cualquier lenguaje de programación específico como C, C++, FORTRAN, VISUAL BASIC, etc. No existe una notación formal o estándar de pseudocódigo, sino que, cada programador puede utilizar la suya propia.

En este sentido el primer paso es determinar los procesos necesarios:

Guardar en dos variables (DIS Y TMPO) los valores de distancia y tiempo respectivamente; aplicar fórmula $VELOCIDAD = DISTANCIA / TIEMPO$; guardar el resultado en la variable VEL; comunicar el resultado velocidad.

Luego plantear el algoritmo en pseudocódigo:

Paso 1: Inicio



Paso 2: Asignar el número 50 a la constante "DIS"

Paso 3: Asignar el número 20 a la constante "TMPO"

Paso 4: Guardar en la constante "VEL" el resultado de $vel = dis / tmpo$

Paso 5: Imprimir el valor de la variable "VEL"

Paso 6: Final

Ahora sí se hará la traducción del algoritmo en Scratch. Para representar estos pasos, empecemos por hacer primero la simulación de movimiento de la persona. Para ello se empieza por seleccionar el personaje a través de la Pestaña "Disfraces" . Luego en Importar , lo que abrirá la siguiente ventana, donde debemos escoger de las diferentes categorías nuestro personaje. En este caso entraremos a people (Personas).



Donde seleccionaremos los cinco archivos llamados boy4-walking-a (b, c, d, e) dando clic en aceptar.



En la pestaña se mostrarán los cinco personajes más dos que por defecto carga Scratch. Estos debemos eliminarlos dando clic en



Importar disfraces

Como se puede observar cada personaje tiene una postura diferente, simulando una parte del movimiento del mismo actor, lo que permitirá crear la ilusión de movimiento. Un sprite o actor es la reunión de todas sus imágenes.

Ahora se elige un escenario para nuestra simulación. Ir a la zona de sprites y escenarios y dar clic en Escenario, luego desde la zona de pestañas elegir “Fondos” y luego “Importar”:




Opción escenario

Entre las opciones, para el caso, elegimos Nature donde seleccionaremos Desert.




Importar fondos



Ahora eliminemos fondo 1, que es el fondo blanco, dando clic en 



Eliminar fondo

Teniendo el fondo y el sprite o actor, vamos a aplicarle movimiento. Para hacerlo se da clic en la pestaña programas   . Luego se elige en la Paleta de Bloques la

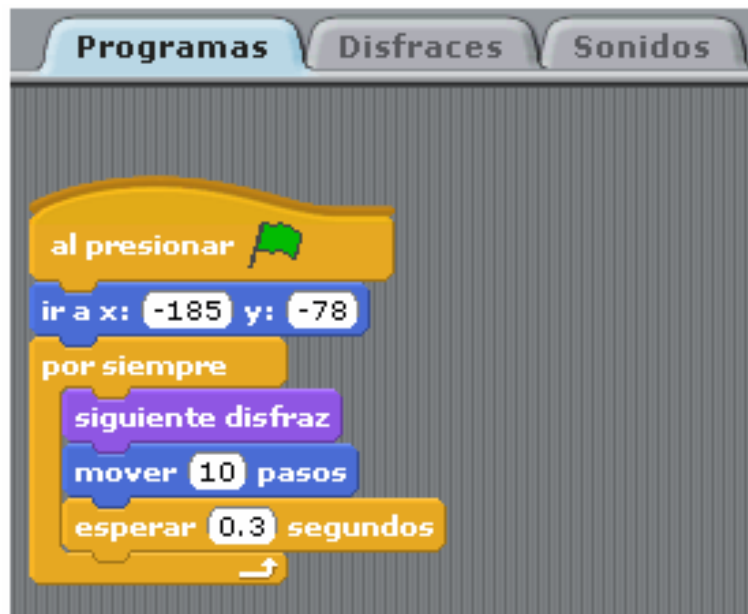
opción Control, y de este grupo el bloque al presionar  arrastrándolo con clic sostenido hasta el Área de Scripts. Esta opción significa que el programa correrá cuando presionemos la bandera verde.

Ahora desde el grupo  se selecciona de la paleta de bloques , asignando un valor en X y Y para indicar que el sprite se posicione siempre en el mismo lugar de inicio. Arrastrarlo debajo del anterior bloque.


Debe verse en el Área de Scripts así:



Regresmos a la opción Control y se busca el bloque Por siempre. Ahora se va a Apariencia y se elige el bloque Siguiete disfraz, encajándolo dentro del bloque anterior, como se ve en la siguiente imagen. Lo mismo se va a hacer con las siguientes instrucciones: desde Movimiento traer el bloque Mover 10 pasos. Por último desde Control se suma el bloque Esperar 1 segundo, asignando el valor 0,3.

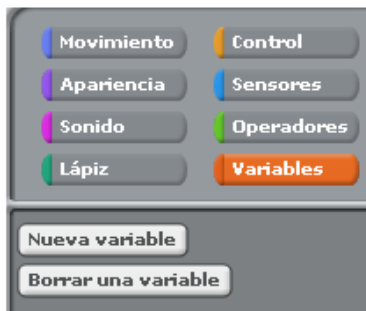


Mover por siempre

Para observar cómo opera este conjunto de bloques de programación, se da clic en la bandera verde. Para detenerlo, clic en el botón rojo . Con esto tenemos el personaje listo para moverse, a fin de simular el ejercicio planteado al inicio.

La segunda parte de nuestro trabajo es entonces calcular la velocidad de una persona que recorre una distancia de 50 m en 20 segundo, usando para ello Scratch. Recuerde que la fórmula que aplica es $VELOCIDAD = DISTANCA / TIEMPO$.

Volvamos a dar clic en la opción control y seleccionemos de la Paleta de Bloques el Control “Al presionar”. Posteriormente desde Variables se selecciona Nueva variable, y en la caja de diálogo que aparece escribimos (DIS) que es el nombre que le daremos a la variable Distancia. Luego haremos lo mismo para crear las otras variables TMPO y VEL. El proceso debe mostrar:




Nueva variable



Nombre de variable




Variables creadas

Debajo del bloque “Al Presionar” ubicado en el Área de Scripts, se arrastra el bloque  y le asignamos el valor 50, el cual representa la distancia que recorrerá nuestro personaje. Arrastramos otro bloque idéntico, pero esta vez se cambia el nombre de la variable que aparece. Se despliegan las opciones a través de la flecha que tiene Dis al lado, observando:



Menú variable

Se elige entre ellas la variable TMPO, asignando el valor 20, con lo que se representan los 20 segundos que tarda la persona en recorrer la distancia de 50 m. Se repite esta acción pero con un bloque que tenga la variable Vel.

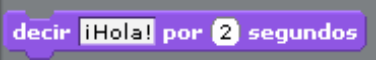
Para el siguiente paso debemos dar clic en la opción Operadores de la Paleta de Bloques, eligiendo el operador de división .



Se encaja dentro del bloque velocidad así: . Se regresa a la opción Variables seleccionando  para ensamblarla en


el bloque anterior . Se repite la acción con la variable Tmpo, para ver:

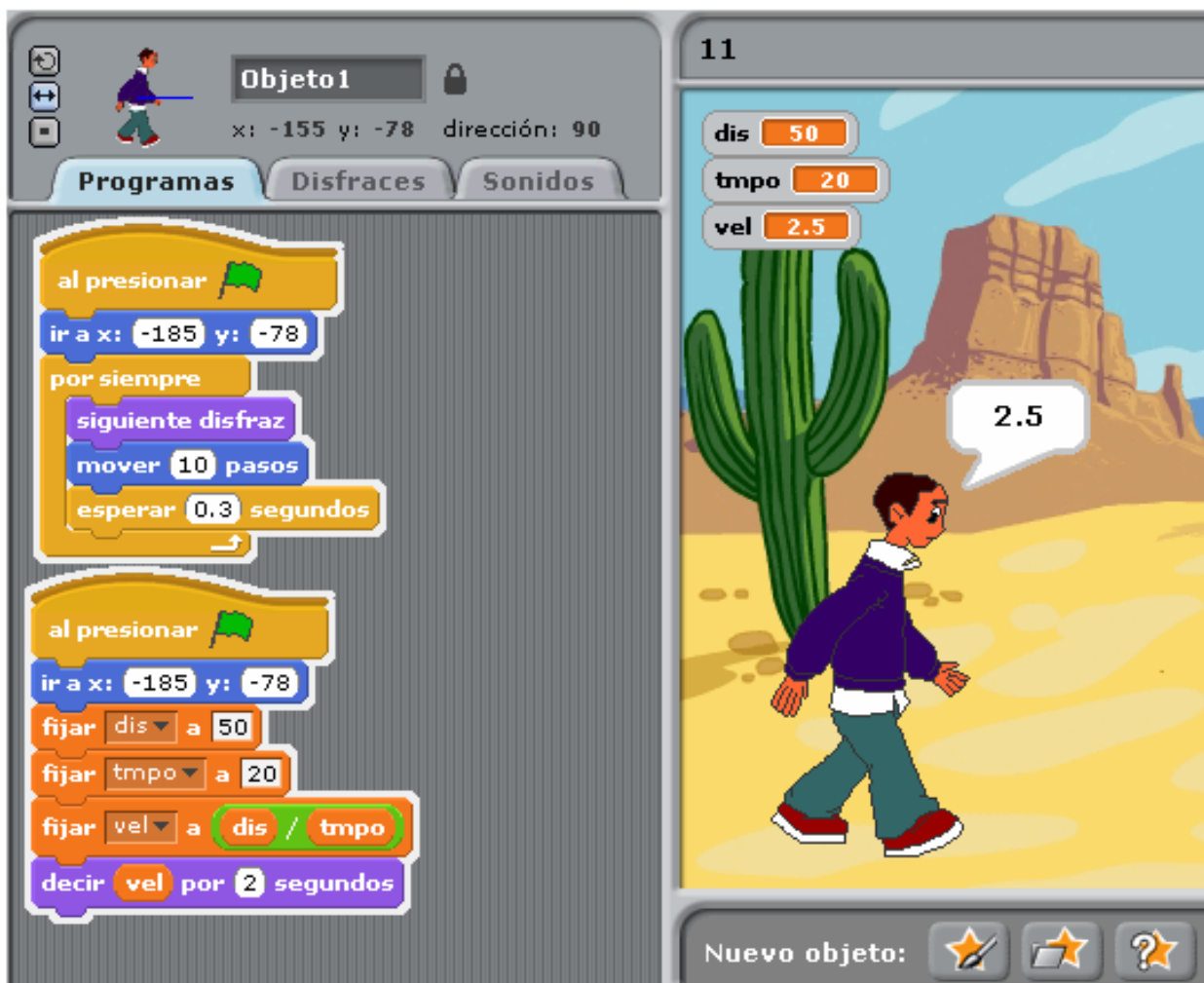


Como puedes observar hemos aplicado la fórmula VELOCIDAD=DISTANCIA/TIEMPO usando el programa SCRATCH.

Para que el sprite nos diga el resultado se elige la opción Apariencia en la Paleta de Bloques, agregando el bloque  en el Área de Scripts. Regresamos a la opción

Variables y se selecciona , encajándola en el bloque anterior .

Si todo marcha bien sólo nos resta dar clic en la bandera verde  para ver los resultados del programa.



Resultado 1

Este es sólo un pequeño y sencillo ejemplo que ayuda a mostrar cómo Scratch ayuda a construir simulaciones, permitiendo que el estudiante comprenda el concepto, su aplicación y utilidad de forma práctica.

Es compromiso de los docentes y las Instituciones Educativas desarrollar estrategias para explotar todo el potencial de esta y otras herramientas al servicio de una educación más pertinente en contexto, a la vez que profundamente relacionada con las posibilidades de interacción global para la construcción colectiva de conocimientos.

Crear y Publicar con las TIC en la escuela

Editores:

Jorge Jair Moreno Chaustre, Sandra Lorena Anaya Díaz,
Ulises Hernandez Pino, Marcela Hernández

Grupo de I+D en Tecnologías de la Información - GTI

Grupo de Investigación en Enseñanza de las Ciencias y Contextos Culturales - GEC

Autores:

Ángela Benavides Maya, Bairon Alvira Manios, Ederson Córdoba Melo, Emilse Patricia Rodríguez,
Ennio Erazo, Greis Silva Calpa, Heidy Valencia Palomino, Heliana Sarria Vivas,
Jairo Montilla Muñoz, Julieth Fajardo Gutierrez, Lilian Cruz Cruz, Omar Trejo Narváez,
Paola Andrea Rosero, Sandra Milena Botina, Stephania Bolaños Muñoz

Universidad del Cauca - Computadores para Educar
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Sistemas

© Universidad del Cauca
© Computadores para Educar
© Los autores

Universidad del Cauca
Calle 5 No. 4 – 70
Conmutador: (57+2) 8 20 98 00
Popayán - Colombia

Computadores para Educar
Carrera 8 entre Calle 12 y 13, Edificio Murillo Toro - Piso 5
Conmutador: (57+1) 3 44 22 58
Bogotá, D. C. - Colombia.

ISBN 978-958-732-083-1
Sello Editorial Universidad del Cauca
1ª Edición: 1100 Ejemplares

Diseño de carátula: Impresora FERIVA S.A.
Fecha: Mayo de 2011



Se permite la copia, presentación pública y distribución de este libro bajo los términos de la Licencia Creative Commons Reconocimiento – No Comercial, la cual establece que en cualquier uso: 1) se de crédito a los autores del libro; 2) no se utilice con fines comerciales; y 3) se den a conocer estos términos de licenciamiento. Conozca la versión completa de esta licencia en la dirección web: <http://creativecommons.org/licenses/by-nc/2.5/co/>

Este documento, al igual que otros generados como parte de la estrategia de formación y acompañamiento de la Universidad del Cauca – Computadores para Educar, se puede descargar a través de la dirección web: <http://www.unicauca.edu.co/cpepacificoamazonia>