

Conceptos de programación desarrollados con Scratch

2012-05-01

Conceptos de programación desarrollados con Scratch

En el proceso de crear con Scratch historias interactivas, juegos y animaciones, niños y jóvenes tienen una excelente oportunidad para desarrollar importantes habilidades de pensamiento y aprender conceptos de computación. Este documento se centra en la enseñanza de conceptos como secuencia, iteración, variables, sentencias condicionales, entradas vía teclado, manejo de eventos, hilos, aleatoriedad y lógica booleana, mediante explicaciones y ejemplos.

En el proceso de crear con Scratch historias interactivas, juegos y animaciones, niños y jóvenes tienen una excelente oportunidad para desarrollar importantes habilidades de pensamiento y aprender conceptos de computación. Este documento se centra en la enseñanza de conceptos como secuencia, iteración, variables, sentencias condicionales, entradas vía teclado, manejo de eventos, hilos, aleatoriedad y lógica booleana, mediante explicaciones y ejemplos.

•

CONCEPTOS DE PROGRAMACIÓN DESARROLLADOS

CON SCRATCH

En el proceso de creación de historias interactivas, juegos y animaciones con Scratch, niños y jóvenes tienen una excelente oportunidad para desarrollar importantes habilidades de pensamiento y para aprender conceptos de computación.

HABILIDADES DE SOLUCIÓN DE PROBLEMAS Y DE DISEÑO

1. Razonamiento lógico
2. Resolver problemas de depuración
3. Desarrollar ideas a partir de una concepción inicial de proyecto
4. Mantener la atención constante y la perseverancia

**

IDEAS FUNDAMENTALES SOBRE PROGRAMACIÓN Y COMPUTADORES**

Los programas indican al computador lo que debe hacer, de manera precisa y paso a paso

- Programar computadores no requiere habilidades especiales, más allá de un pensamiento claro y metódico

CONCEPTOS ESPECÍFICOS DE PROGRAMACIÓN:

|

CONCEPTO

|

EXPLICACIÓN

|

EJEMPLO [1]****

|

|

Secuencia

|

Para crear un programa en Scratch, se requiere pensar sistemáticamente sobre el orden de los pasos.

|



|

|

Iteración (ciclos)

|

por siempre y **repetir** se utilizan para crear iteraciones (repetición de una serie de instrucciones).

|



|

|

Variables

|

Las variables sirven para almacenar números o cadenas de caracteres (palabras). Las instrucciones correspondientes a **variables** permiten crearlas y usarlas en un programa. Scratch admite tanto variables globales (para todos los objetos) como específicas para un solo objeto.

|



|

|

Sentencias Condicionales

|

si y **si-no** verifican si una proposición simple o compuesta es verdadera (si se cumple una condición).

|



|

|

Entradas vía teclado

|

preguntar y esperar solicita a los usuarios escribir algo mediante el teclado.

respuesta es una variable que almacena lo último que se ingresó vía teclado.



|

|

Manejo de eventos

|

al presionar tecla y **al presionar objeto** son ejemplos del manejo de eventos. Estas instrucciones de control responden a eventos provocados por el usuario o por otra parte del programa.



|

|

Hilos (paralelismo)

|

Poner en marcha dos pilas de instrucciones al mismo tiempo hace que se creen dos hilos independientes que se ejecutan en paralelo.

|



|

|

Números al azar

|

La instrucción **número al azar entre** selecciona un número entero dentro de un rango dado.

|



|

|

Lógica booleana

|

Las proposiciones compuestas se forman con dos o más proposiciones sencillas unidas por operadores lógicos (**y**, **o**, **no**). Por ejemplo: evaluar la condición “variable lados tiene almacenado un valor menor o igual a 9”.



|

Diseño de interfaz de usuario

|

Con Scratch se diseñan interfaces de usuario interactivas. Por ejemplo, usar objetos para que funcionen como botones. Por ejemplo, al hacer clic sobre el objeto “Lápiz” se ejecuta un conjunto de instrucciones.



|

Coordinación y sincronización

|

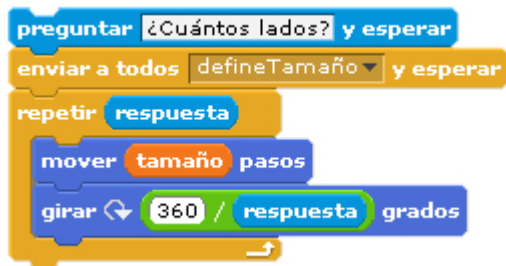
La instrucción **enviar a todos** manda un mensaje a todos los Objetos y espera a que se ejecuten las acciones de los Objetos activados.

La instrucción **al recibir** coordina acciones de diferentes objetos.

Este par de instrucciones permiten la sincronización.

|

Por ejemplo, cuando un objeto “envía a todos defineTamaño”...



...entonces se ejecutan las instrucciones debajo de la instrucción “al recibir defineTamaño”



|

|

Listas (arreglos)

|

Las Listas son un tipo de estructura de datos que puede considerarse un arreglo bidimensional de “n x 1”. Con varias listas se puede conformar una matriz (arreglo bidimensional de n x m).

Las instrucciones correspondientes a **listas** permiten almacenar y acceder arreglos de números o cadenas de caracteres.

|



|

|

Interacción dinámica

|

Las instrucciones “**x del ratón**”, “**y del ratón**” y “**volumen del sonido**” se pueden utilizar como entrada dinámica para interactuar en tiempo real con los programas de Scratch.

|



|

NOTAS DEL EDITOR:

[1] Con el fin de darle mayor coherencia a los ejemplos de los diferentes conceptos, se cambiaron los propuestos en el documento original en inglés por otros que giran en torno al dibujo de polígonos regulares.

CRÉDITOS:

Traducción al español por Eduteka del documento

<http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/files/file/ScratchProgrammingConceptsv14.pdf> “Programming Concepts and Skills Supported in Scratch”, elaborado por el

grupo de investigación del MIT Media Lab, “Lifelong Kindergarten” (Jardín Infantil para toda la vida).



La traducción al español de este documento la realizó Eduteka con el apoyo de [Motorola Solutions Foundation](#) y de la ONG [Give to Colombia](#).

|

*Publicación de este documento en EDUTEKA: Mayo 01 de 2012.

Última modificación de este documento: Mayo 01 de 2012.*