

***Desarrollo del
pensamiento
computacional mediante
Scratch en estudiantes de
educación media del
municipio de Pamplona.
Caso de estudio: colegio
Brighton***

Autor

Nelson Andres Acevedo Mera

**Desarrollo del pensamiento computacional mediante
Scratch en estudiantes de educación media del municipio
de Pamplona**

Nelson Andres Acevedo Mera

**Universidad de Pamplona
Facultad de Ingeniería y Arquitectura
2018**

**Desarrollo del pensamiento computacional mediante
Scratch en estudiantes de educación media del municipio
de Pamplona**

Nelson Andres Acevedo Mera

**Trabajo de grado para optar el título de ingeniero de
sistemas**

Director

Dr. José Orlado Maldonado Bautista

Co-Director

Mc. Edgar Alexis Albornoz Espinel

**Universidad de Pamplona
Facultad de Ingeniería y Arquitectura
2018**

DEDICATORIA

*A Dios,
Mis padres Gloria y Nelson,
Y a mi abuela María Elena.*

AGRADECIMIENTOS

Quiero dar mis más sinceros agradecimientos a Dios por darme la fortaleza en los momentos difíciles, a mi familia en especial a mis padres, hermanas y mi abuela materna doña Elena, dado que sin ellos no nunca hubiese logrado llegar tan lejos en mis estudios, me dieron apoyo incondicional en los momentos difíciles y nunca dejaron que me sintiera solo ni derrotado.

Agradezco enormemente a Alba Jáuregui “mi amor” y a su familia, quienes me apoyaron de todas las formas posibles, y se convirtieron en una segunda familia que siempre me brinda un espacio donde sentirme seguro y tranquilo.

A Esteban Y Dayana quienes a pesar de ser tan pequeños y no comprender aun la magnitud de este logro, me alegran en los momentos difíciles y me ayudan a ver el mundo con la creatividad e imaginación que aún conservan los niños.

Por ultimo pero no menos importante a mis profesores que a lo largo de mi vida me han ayudado a formar como una persona integral desde los grados de primaria, es especial agradezco a los profesores Orlando Maldonado y Edgar Albornoz por guiarme y motivarme en el desarrollo de este trabajo de grado, gracias a su invaluable ayuda este proyecto se logró culminar de la mejor manera.

Muchas gracias a todos.

CONTENIDO

CAPITULO I	16
INTRODUCCION	16
1.1. JUSTIFICACIÓN.....	16
1.2. PLANTEAMIENTO DEL PROBLEMA	17
1.3. OBJETIVO GENERAL.....	18
1.4. OBJETIVOS ESPECIFICOS.....	18
CAPITULO II	19
MARCO TEORICO	19
2.1. Pensamiento computacional.....	19
2.2. LOGO	25
2.3. Solución de problemas computacionales	26
2.3.1. Proceso de pensamiento.....	29
2.4. Scratch.....	30
2.5. Institución educativa Bethlemitas Brighton	35
CAPITULO III	37
ESTADO DEL ARTE	37
3.1. Nivel internacional	37
3.2. Nivel nacional	43
3.3. Nivel Local	53
CAPITULO IV	54
METODOLOGÍA	54
4.1. Descripción de la metodología.....	54
4.2. Población-muestra	55
4.3. Herramientas utilizadas para la recolección de datos.	56
4.4. Recolección de datos.....	57
4.5. El análisis de los datos	57
CAPITULO V	59
DESARROLLO DE LAS ACTIVIDADES	59
5.1. Diseñar encuesta de predisposición para el aprendizaje de la programación de computadores	59
5.2. Aplicar la encuesta a los estudiantes de colegio.....	61
5.3. Tabular resultados de la encuesta	61

5.4.	Diseñar la prueba técnica inicial y la prueba técnica final para medir el pensamiento computacional.....	68
5.5.	Aplicación de la prueba técnica inicial	93
5.6.	Tabulación de los resultados de la prueba técnica inicial	94
5.7.	Asistencia al curso como observador y registro del comportamiento de los estudiantes 97	
5.8.	Tabulación de resultados de la prueba técnica final.....	105
5.9.	Comparación de los resultados obtenidos en la prueba técnica inicial y la prueba técnica final	108
CAPITULO VI		111
CONCLUSIONES Y TRABAJOS FUTUROS		111
6.1.	Conclusiones.....	111
6.2.	Trabajos Futuros.....	112
REFERENCIAS		113
ANEXOS		116

Resumen

A nivel mundial los programas de formación académicos basados en las tecnologías de la información, han visto como el número de personas interesadas en ingresar a dichos programas ha disminuido drásticamente, esta situación ha llevado a que se cree un déficit de profesionales en dichas áreas, los motivos por los cuales los estudiantes que terminan el bachillerato cada vez muestran menos interés por ingresar a dichos programas son muy variados, abarcan aspectos personales como las dificultades económicas y familiares, hasta otros de carácter netamente académico, como temor por las matemáticas y la física.

El programa de Ingeniería de Sistemas no es ajeno a esta problemática, y adicional a el bajo número de personas inscritas a comparación de otros programas se suman las altas tasas de deserción y mortalidad académica presentes en la carrera, se ha detectado que las falencias presentadas en el área de la programación han resultado determinantes a la hora del estudiante tomar la decisión de continuar o no en el programa de formación académica.

El proceso de aprender a programar es complejo e involucra mucho tiempo y dedicación por parte del interesado, aunque algunas personas presentan facilidad para desarrollar las habilidades mentales para construir algoritmos, a las personas en general les cuesta trabajo comenzar el desarrollo de dichas habilidades, y en algunos casos esto puede llevar a que una persona que no posee la facilidad para general algoritmos, decida alejarse por completo del mundo de la programación e iniciar sus estudios en otras áreas diferentes.

Buscando una solución al número de personas que desertan del programa de Ingeniería de Sistemas de la Universidad de Pamplona, se desarrolla este proyecto que se acercó a los estudiantes de 11° de un colegio del municipio de Pamplona con la finalidad de estimular en estos el interés de ingresar a los diferentes programas relacionados con las tecnologías de la información que oferta la Universidad de Pamplona, en especial a el programa de Ingeniería de Sistemas. Para lograr dicho objetivo se elaboró un curso de programación en Scratch, que tenía varios objetivos:

- Romper el paradigma tradicional acerca de lo que se piensa sobre las funciones y conocimientos que desempeña un ingeniero de sistemas.
- Estimular el desarrollo de las habilidades propias del pensamiento computacional en los estudiantes a través del estudio de la programación.
- Fomentar el ingreso al programa de Ingeniería de Sistemas de la Universidad de Pamplona, mediante la demostración de lo que se puede llegar a desarrollar con la ayuda de la programación.
- Buscar un perfil para la selección de jóvenes talentos, que puedan aportar al programa futuros buenos estudiantes que representen la institución en eventos o con buenas calificaciones en pruebas externas como las ECAES.

La metodología de investigación del proyecto es de tipo cualitativa dado que se centra en la recolección y análisis de datos para responder las preguntas de investigación planteadas en el proyecto.

El tipo de muestra según el tipo de estudio con el cual se está realizando la investigación es en un grupo de 25 personas. Para la recolección de datos se han realizado una serie de pruebas técnicas que se han aplicado con el uso de las tecnologías de información, también mediante la observación directa ha sido posible registrar la actividad observada a lo largo del desarrollo del proyecto cuando se presenta contacto con la muestra seleccionada, se utilizan encuestas a los miembros de la muestra con el fin de agrupar los diferentes perfiles de los estudiantes y comparar como estos afectan el desarrollo del pensamiento computacional.

Con la realización del proyecto se buscó que las habilidades de pensamiento computacional como:

- Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos.
- Organizar datos de manera lógica y analizarlos.
- Representar datos mediante abstracciones, como modelos y simulaciones.
- Automatizar soluciones mediante pensamiento algorítmico.

- Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.
- Generalizar y transferir ese proceso de solución de problemas a una gran diversidad de estos.

Se vean favorecidas, es decir se note un desarrollo en estas al comparar los resultados de las dos pruebas técnicas (inicial y final) presentadas por los estudiantes.

Las etapas bajo las cuales se desarrolló el proyecto fueron:

1. Se realizó toda la documentación necesaria sobre la creación de pruebas técnicas que permitieran medir de manera objetiva las habilidades de pensamiento computacional de los estudiantes.
2. Se crearon las pruebas técnicas (inicial y final) basadas en la consulta previa realizada.
3. Se aplicó la prueba técnica inicial a los estudiantes de 11° seleccionados y a el otro grupo de 11° que no recibiría el curso con el fin de que este segundo grupo validara el grado de dificultad de la prueba haciendo las veces de grupo de control.
4. Se crearon las guías de trabajo bajo las cuales se desarrollaron cada una de las sesiones con los estudiantes, en cada una de ellas se buscaba no solo aprender sobre cierto tema de la programación o de manejo de la herramienta Scratch sino estimulara determinada habilidad del pensamiento computacional.
5. Se dictaron cada una de las sesiones programadas, con la ayuda de estudiantes de trabajo social quienes orientaron el desarrollo de las guías de trabajo con el fin de que el investigador pueda tomar nota del desarrollo de las mismas sin ser un actor participante.
6. Se aplicó la prueba técnica final a los dos grupos de 11°.
7. Se tabularon los resultados de las 2 pruebas técnicas y se compararon entre sí, con el fin de observar como cada una de las habilidades evaluadas se mantuvo igual o si por el contrario se desarrolló.

LISTA DE TABLAS

Tabla 1: Habilidades del pensamiento computacional y acciones desarrolladoras.....	22
Tabla 2: pregunta 2 prueba técnica final. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.....	71
Tabla 3: pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	72
Tabla 4: pregunta 1 prueba técnica final. Organizar datos de manera lógica y analizarlos	76
Tabla 5: Pregunta 1 prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico	80
Tabla 6: Pregunta 2 prioridad, precio y utilidad; prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico.....	81
Tabla 7: Pregunta 2 útiles escolares prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico	81
Tabla 8: pregunta 2 prueba final. Formular problemas de manera que permitan usar computadoras y otras; Representar datos mediante abstracciones, como modelos y simulaciones	85

LISTA DE GRAFICAS

Ilustración 1: Ejemplo de aplicación en LOGO	26
Ilustración 2: Áreas y etapas de la resolución de problemas.....	28
Ilustración 3: Esquema simplificado del Proceso de Solución de Problemas (Gick, 1986)	30
Ilustración 4: Entorno inicial de Scratch versión Offline	31
Ilustración 5: Entorno inicial de Scratch versión Online	32
Ilustración 6: Categorías de instrucciones en Scratch.....	33
Ilustración 7: Entorno inicial de Scratch JR	34
Ilustración 8: Entrada original Institución educativa Bethlemitas Brighton	36
Ilustración 9: Tecnologías usadas grupo experimental.....	61
Ilustración 10: Tecnologías usadas grupo control.....	61
Ilustración 11:Curiosidad por la creación de tecnologías experimental.....	62
Ilustración 12:Curiosidad por la creación de tecnologías grupo control	62
Ilustración 13: Interés por aprender a desarrollar nuevas tecnologías grupo experimental	63
Ilustración 14:Interes por aprender a desarrollar tecnologías grupo de control	63
Ilustración 15: Tecnologías que le gustaría aprender grupo experimental	64
Ilustración 16: Tecnologías que le gustaría aprender grupo de control	64
Ilustración 17:Facilidad o interés por matemáticas y/o lógica matemáticas.....	65
Ilustración 18:Facilidad o interés por matemáticas y/o lógica matemáticas.....	65
Ilustración 19: Carreas a estudias grupo experimental	66
Ilustración 20: carreras a estudiar grupos de control	66
Ilustración 21: Perfil de ingeniero de sistemas según grupo experimental.....	67
Ilustración 22: Perfil de ingeniero de sistemas según grupo	67
Ilustración 23:Imagen de referencia pregunta 1 técnica inicial. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva	68
Ilustración 24:Imagen de referencia pregunta 2 prueba técnica inicial. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.....	69
Ilustración 25 imagen de referencia prueba técnica final. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva	70
Ilustración 26: Opción de respuesta a pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	73
Ilustración 27: Opción de respuesta b pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	73
Ilustración 28: Opción de respuesta c pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	74
Ilustración 29: Opción de respuesta d pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	74
Ilustración 30: imagen e ilustración pregunta 2 prueba técnica inicial. Organizar datos de manera lógica y analizarlos	75
Ilustración 31: Pregunta 2prueba técnica final. Organizar datos de manera lógica y analizarlos	77

Ilustración 32: imagen de referencia pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	86
Ilustración 33: opción a pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	87
Ilustración 34: opción b pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	87
Ilustración 35: opción c pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	88
Ilustración 36: imagen de referencia pregunta 1, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	89
Ilustración 37: imagen de referencia pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	90
Ilustración 38: Opción a, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	91
Ilustración 39: Opción b, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	91
Ilustración 40: Opción c, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos	92
Ilustración 41: Resultados prueba inicial del grupo experimental, por habilidades, en porcentajes	95
Ilustración 42: Resultados prueba inicial del grupo de control, por habilidades, en porcentajes....	95
Ilustración 43: mediciones del curso.....	104
Ilustración 44: Resultados prueba final del grupo experimental, por habilidades, en porcentajes	106
Ilustración 45: Resultados prueba final del grupo experimental, por habilidades, en porcentajes	106
Ilustración 46: Evolución de las habilidades antes y después del curso	110

TABLA DE ANEXOS

Anexo 1.....	116
Anexo 2.....	125
Anexo 3.....	135
Anexo 4.....	147
Anexo 5.....	153
Anexo 6.....	164

CAPITULO I

INTRODUCCION

1.1. JUSTIFICACIÓN

Este proyecto se desarrolló para determinar el impacto que puede tener sobre estudiantes de los últimos grados de los colegios de la ciudad de Pamplona la realización de actividades que estimulen el desarrollo de su pensamiento computacional.

Con el desarrollo del proyecto se buscó comprobar si los estudiantes que lo tomen mejoran sus habilidades lógicas, viendo favorecido su desempeño académico y encontrando en la programación de computadores una herramienta poderosa para mejorar su pensamiento computacional.

Mediante las actividades planeadas se puede incentivar el ingreso de nuevos estudiantes a programas relacionados con las TIC, con una orientación vocacional previa, con el conocimiento de las habilidades y aptitudes requeridas para el éxito académico contribuyendo a reducir la brecha entre la oferta y demanda de profesionales en estas áreas.

Los resultados del proyecto validan una estrategia para la búsqueda de jóvenes talentos que lleguen a ser potenciales estudiantes de carreras relacionadas con las TIC, como el programa de Ingeniería de Sistemas, de tal forma que pueda reducirse las tasas de deserción y mortalidad en los mismos.

1.2. PLANTEAMIENTO DEL PROBLEMA

El programa de Ingeniería de Sistemas de la Universidad de Pamplona tiene una alta tasa de deserción académica, en el último registro de la oficina de planeación de la Universidad e Pamplona se encontró que para el segundo semestre de 2014 la tasa de deserción académica era de 10,6%, lo cual representa una alta cifra, comparada con otros programas académicos que también son ofertados por la Universidad de Pamplona.

(“Oficina de Planeacion - Universidad de Pamplona - Sistemas de Información”, s/f)

La mayoría de personas que ingresan como nuevas al programa de ingeniería de sistemas muchas veces desconocen en que consiste este, así como el perfil laboral que posee el Ingeniero de Sistemas de la Universidad de Pamplona. Se observa en algunos de los jóvenes que se encuentran próximos a graduarse del colegio, potenciales talentos que pueden aportar calidad y un alto nivel académico a el programa, por tanto, se hace imperante la necesidad de incentivar el ingreso de estos jóvenes a programas relacionados las tecnologías de la información y la comunicación (TIC).

Las dificultades en la resolución de problemas es sin duda uno de los más grandes desafíos a los cuales se puede enfrentar cualquier persona, poseer habilidades o desarrollar estrategias para la resolución de problemas es una característica que es difícil de obtener y no todas las personas pueden llegar a desarrollarlas, el pensamiento computacional consiste en desarrollar una serie de habilidades que son muy útiles a la hora de resolver problemas de distinta índole, no solo de problemas relacionados con la computación.

Una de las estrategias más aceptadas a nivel mundial para el desarrollo del pensamiento computacional es el aprendizaje de la programación de computadores, sin importar el lenguaje que se emplee para aprender este.

Teniendo en cuenta lo anterior se buscó medir el impacto que tiene en el desarrollo del pensamiento computacional la aplicación de un curso de introducción a la programación de computadores en Scratch que aparte de buscar desarrollar el pensamiento computacional

de los estudiantes de colegio, motive a estos a ingresar en el programa de Ingeniería de Sistemas de la Universidad de Pamplona.

1.3.OBJETIVO GENERAL

Determinar el impacto obtenido con el desarrollo de un curso de Scratch en el desarrollo del pensamiento computacional, en estudiantes del grado 11°, mediante la aplicación de pruebas técnicas y observaciones sobre campo.

1.4.OBJETIVOS ESPECIFICOS

- Evaluar la predisposición para el aprendizaje de la programación de computadores en los estudiantes de grado 11° del colegio Betlemitas Brighton.
- Determinar las habilidades en la resolución de problemas computacionales de los participantes del curso, antes del desarrollo del mismo.
- Analizar el desempeño de los estudiantes durante el desarrollo del curso de introducción a la programación, diseñado en Scratch con actividades previamente seleccionadas y preparadas.
- Determinar mediante la aplicación de pruebas posteriores, el impacto alcanzado con el desarrollo del curso en el mejoramiento de las habilidades para la solución de problemas computacionales

CAPITULO II

MARCO TEORICO

2.1. Pensamiento computacional

Según Jeannette Wing vicepresidente corporativo de Microsoft Research y profesora de Computer Science Department Carnegie Mellon University, El pensamiento computacional consiste en la resolución de problemas, el diseño de los sistemas, y la comprensión de la conducta humana haciendo uso de los conceptos fundamentales de la informática.

(Zapata-Ros, 2015)

Se resalta algunos rasgos muy útiles para establecer un corpus curricular para el aprendizaje basado en el pensamiento computacional.

- **En el pensamiento computacional se conceptualiza, no se programa:** Es preciso pensar como un científico de la computación. Se requiere un pensamiento en múltiples niveles de abstracción.
- **En el pensamiento computacional son fundamentales las habilidades no memorísticas o no mecánicas:** Memoria significa mecánico, aburrido, rutinario. Para programar los computadores hace falta una mente imaginativa e inteligente. Hace falta la emoción de la creatividad. Esto es muy parecido al pensamiento divergente.
- **En el pensamiento computacional se complementa y se combina el pensamiento matemático con la ingeniería:** Ya que, al igual que todas las ciencias, la computación tiene sus fundamentos formales en las matemáticas. La ingeniería nos proporciona la filosofía base de que construimos sistemas que interactúan con el mundo real.

En el pensamiento computacional lo importante son las ideas, no los artefactos: Quedan descartados por tanto la fascinación y los espejismos por las novedades tecnológicas. Y mucho menos estos factores como elementos determinantes de la resolución de problemas o de la elección de caminos para resolverlos. (Wing, 2006)

El Pensamiento Computacional es un proceso de solución de problemas que incluye (pero no se limita a) las siguientes características:

- Formular problemas de manera que permitan usar computadores y otras herramientas para solucionarlos
- Organizar datos de manera lógica y analizarlos
- Representar datos mediante abstracciones, como modelos y simulaciones
- Automatizar soluciones mediante pensamiento algorítmico (una serie de pasos ordenados)
- Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva
- Generalizar y transferir ese proceso de solución de problemas a una gran diversidad de estos

Estas habilidades se apoyan y acrecientan mediante una serie de disposiciones o actitudes que son dimensiones esenciales del Pensamiento Computacional. Estas disposiciones o actitudes incluyen:

- Confianza en el manejo de la complejidad
- Persistencia al trabajar con problemas difíciles
- Tolerancia a la ambigüedad
- Habilidad para lidiar con problemas no estructurados
- Habilidad para comunicarse y trabajar con otros para alcanzar una meta o solución común (Gonz, 2013)

El concepto de pensamiento computacional es una competencia compleja relacionada con un modelo de conceptualización específica de los seres humanos que desarrolla ideas y vinculada con el pensamiento abstracto-matemático y con el pragmático-ingenieril que se aplica en múltiples aspectos de nuestra vida diaria. El pensamiento computacional no es sinónimo de capacidad para programar un ordenador, puesto que requiere pensar en diferentes niveles de abstracción y es independiente de los dispositivos. Se puede desarrollar pensamiento computacional sin utilizar ordenadores (basta papel y lápiz), si bien los dispositivos digitales nos permiten abordar problemas que sin ellos no nos atreveríamos a enfrentar. Por otra parte, es una competencia básica que todo ciudadano debería conocer para desenvolverse en la sociedad digital, pero no es una habilidad «rutinaria» o «mecánica», ya que es una forma de resolver problemas de manera inteligente e imaginativa (cualidades humanas que no poseen los ordenadores). Además, posee las características de combinar abstracción y pragmatismo, puesto que se fundamenta en las Matemáticas, un mundo de ideas, y se desarrolla a partir de proyectos de ingeniería que interactúan con el mundo real. Los conceptos computacionales se utilizan para enfocar y resolver problemas reales, comunicarnos con otras personas y gestionar múltiples aspectos de nuestra vida cotidiana.

(Valverde-Berrocoso, Fernández-Sánchez, & Garrido-Arroyo, 2015)

Para ayudar al desarrollo de las habilidades descritas anteriormente una de las herramientas usadas más ampliamente es el aprendizaje de la programación de computadores, en ambientes escolares se utilizan los lenguajes de alto nivel llamados lenguajes de programación basados en un enfoque pedagógico.

(Urbina Ramírez, 1999)

Son lenguajes de programación que se desarrollan y utilizan con el fin de brindar un enfoque pedagógico que permita y facilite el aprendizaje en diferentes áreas del conocimiento como matemáticas, cibernética entre otras de una manera innovadora y didáctica, en la actualidad Scratch es uno de los lenguajes de programación con enfoque pedagógico más difundidos y utilizadas a nivel mundial, antes de Scratch fuera creado, se desarrolló LOGO,

uno de los lenguajes de programación pioneros en el área de la pedagogía en una época donde se creía que la programación era de uso exclusivo de los grandes desarrolladores de software y solo podía ser utilizada a través de complejos y largos códigos que se utilizaban para crear programas de software complejos y costosos.

Tabla 1: Habilidades del pensamiento computacional y acciones desarrolladoras

<i>Habilidad del pensamiento computacional a desarrollar</i>	<i>Conocimientos que ayudan al desarrollo de la habilidad</i>
Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos	El desarrollo de esta habilidad se facilita mediante el estudio y aprendizaje de creación de pseudocódigos y diagramas de flujo, debido a que los pseudocódigos y los diagramas de flujo son representaciones abstractas de programas de computador, estos ayudan al aprendiz a entender la forma en la cual un computador recibe, procesa y transmite la información para el usuario así como la transmisión de la misma hacia otros procesos del mismo programa u otros dispositivos.
Organizar datos de manera lógica y analizarlos	El uso de variables y condicionales son aquellas acciones que más ayudan al desarrollo de esta habilidad, debido a que cuando se comprende y utiliza el manejo de variables se puede hacer uso de estas para clasificar y depurar información de una manera más ordenada y sencilla, por otra parte el manejo de condicional ayuda a entender la importancia que pueda tener el valor de algún dato en un determinado

	<p>momento, permitiendo comprender que según este varié se puede o no ejecutar una u otra operación.</p>
<p>Representar datos mediante abstracciones, como modelos y simulaciones</p>	<p>Al igual que en la habilidad de Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos, el uso de pseudocódigos y diagramas de flujo, resulta muy útil para que los alumnos observen y desarrollen con mayor facilidad el uso de representaciones abstractas de la realidad, a su vez brindan una base inicial para que los alumnos puedan simular lo que puede llegar a ocurrir en el interior de un programa de computador cuando a este se le somete a determinados escenarios mediante la implementación de pruebas de escritorio que permitan corroborar el funcionamiento correcto de un determinado algoritmo.</p>
<p>Automatizar soluciones mediante pensamiento algorítmico</p>	<p>Mediante el manejo de ciclos repetitivos y solicitud de datos por teclado, es posible que el alumno comprenda y asimile que algunos procesos que deben realizarse varias veces pueden ser definidos e implementados como ciclos repetitivos.</p> <p>El estudiante cuando solicita datos por teclado dentro de un ciclo repetitivo, no</p>

	<p>solo comprende la ejecución de un proceso varias veces, sino que también observa que sin importar las entradas o salidas de un proceso este puede llegar a ser generalizado y ejecutado como se hace con los procesos similares entre sí.</p> <p>El razonamiento numérico, ayuda en gran medida al desarrollo del pensamiento algorítmico,</p>
<p>Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva</p>	<p>Con el uso de If anidados y Switch, es posible observar y ajustar un el comportamiento de un programa de computador según cambien los parámetros que este requiere, lo que ayuda a que el estudiante vea que recursos o secuencias son necesarios en la solución de un problema, de esta forma se optimizan los recursos usando solo los necesarios.</p>
<p>Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos</p>	<p>Mediante la replicación de ejemplos en ejercicios similares con diferentes grados de dificultad, el estudiante puede afianzar el manejo de un tema u ver cómo es posible generalizar y adaptar una solución a diferentes problemas del mismo tipo.</p>

2.2. LOGO

Logo es un lenguaje de programación que nace en los años 60. Diseñado con fines didácticos por Danny Bobrow, Wally Feurzeig y el matemático Seymour Papert, que creo la primera versión de este lenguaje en el año 1967. El LOGO se basa en las características del lenguaje LISP. A pesar de que Logo no fue creado con la finalidad de usarlo para enseñar programación, puede usarse para enseñar la mayoría de los principales conceptos de la programación, ya que proporciona soporte para manejo de listas, archivos y entrada/salida.

Este lenguaje no está limitado a un tópico en particular o a una materia específica. Normalmente se puede realizar programas con el lenguaje LOGO sobre: matemáticas, lenguaje, música, robótica, telecomunicaciones y ciencias.

También se usa para simulaciones, presentaciones multimedia, etc. Sin embargo, es más comúnmente utilizado para la exploración de las Matemáticas ya que los gráficos de tortuga de LOGO proveen un ambiente naturalmente matemático.

Debido a que la tortuga se mueve una determinada distancia y gira un número dado de grados, el estudio de geometría mediante la construcción e investigación de polígonos y figuras hace de LOGO una herramienta de aprendizaje poderosa.

Fundamentalmente consiste en presentar a los niños retos intelectuales que puedan ser resueltos mediante el desarrollo de programas en LOGO. El proceso de revisión manual de los errores contribuye a que el niño desarrolle habilidades metacognitivas al poner en práctica procesos de autocorrección.

Otra gran cualidad del LOGO es que a este lenguaje se le pueden enseñar nuevos comandos y a su vez crear otros a partir de estos nuevos. Por esta razón se puede decir que el LOGO tiene la cualidad de ser extensible. Es un lenguaje poderoso en el sentido que es capaz de crear programas desde los más simples a programas complejos.

(Carro, Dpto, & Villamil, 2009)

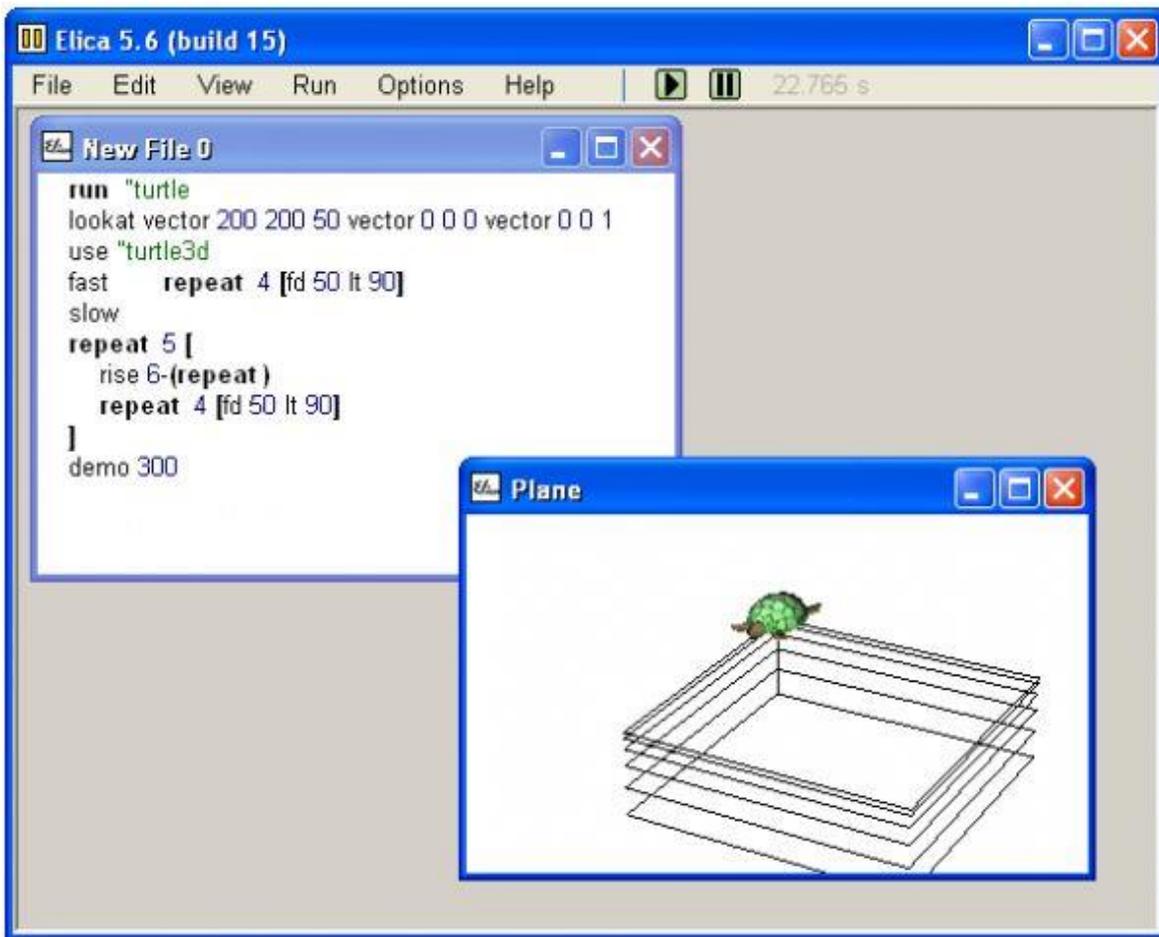


Ilustración 1: Ejemplo de aplicación en LOGO

Fuente: tomado de <http://www.fotoseimagenes.net/lenguaje-de-programacion-logo>

2.3. Solución de problemas computacionales

En la actualidad todas las personas en su día a día se encuentran con problemas, pero la diferencia se encuentra en que hay personas que encuentran soluciones muy fácilmente y otras que tienen más dificultades a la hora de hacerlo, porque estos no poseen las mismas habilidades del pensamiento y solución de problemas necesarias para encontrar de manera rápida una solución a los inconvenientes.

La resolución de problemas es el proceso a través del cual podemos reconocer las señales que identifican la presencia de una dificultad, anomalía o entorpecimiento del desarrollo

normal de una tarea, recolectar la información necesaria para resolver los problemas detectados y escoger e implementar las mejores alternativas de solución.

(Del, Aprendizaje, López, & Olmo, s/f)

En este momento uno de los grandes objetivos que tienen la educación es adquirir estas habilidades para los alumnos, aunque no se han desarrollado los procesos necesarios para lograr este objetivo en las instituciones.

La metodología general de resolución de problemas es un método que permite no solamente resolver un problema puntual, sino que también deberá ser vista como una estrategia explícita que permite crear, adquirir y transferir nuevos conocimientos.

(IIPE Buenos Aires & Ministerio de educación de la Nación - Argentina, 2000)

Lo importante es lograr generar en los alumnos cada etapa de procesos de resolución, el cambio de cada individuo en su comportamiento, el aprender es comprender los problemas de nuevas formas y verlos diferente a los demás, ayudando a fortalecer la resolución de los mismos.

La resolución tiene tres desafíos que son:

- La comprensión del problema.
- La creación de una estrategia de resolución o intervención.
- El logro del mejoramiento o la solución al problema.

(IIPE Buenos Aires & Ministerio de educación de la Nación - Argentina, 2000)

Para resolver dichos desafíos se encuentran siete etapas para la solución de problemas que se presenta sucesivamente, aunque hay ocasiones en donde hay que retroceder estas son:

- **Identificar el problema:** se ocupa de estudiar las manifestaciones visibles del problema, conocerlo a través de indicadores y registros, "cercarlo" en sus efectos, delimitar el área y población de influencia, cuantificarlo, compararlo con la situación existente en otras regiones o provincias.

- **Explicar el problema:** Explicar el modelo supone indagar sobre las causas que lo componen, requiere una recopilación de información y una organización conceptual de la información que permitan ver causas
- **Idear las estrategias alternativas:** Esta fase es eminentemente creativa, aunque tiene su punto de partida en el modelo explicativo elaborado
- **Decidir la estrategia:** La mejor estrategia es aquella que relaciona el problema a resolver.
- **Diseñar la intervención:** es la programación cuidadosa y minuciosa de todas las acciones, roles, recursos, decisiones auxiliares, plazos, instrumentos, métodos y asesoramientos necesarios para llevar adelante el proceso de mejoramiento.
- **Desarrollar la intervención:** es entendido aquí como la implementación.
- **Evaluar los logros:** la evaluación exige un diseño específico que establezca cuáles serán sus objetivos, las técnicas de relevamiento de datos, los tiempos que durará la evaluación.

Objetivos	Etapas
<p>COMPRENDER EL PROBLEMA en su complejidad y en su resonancia para los grandes objetivos educativos.</p>	<p>1. Identificar el problema. 2. Explicar el problema.</p>
<p>CREAR UNA ESTRATEGIA DE SOLUCIÓN apoyada en las fortalezas, que minimice los efectos negativos y que asegure logros reales.</p>	<p>3. Idear las estrategias alternativas. 4. Decidir la estrategia. 5. Diseñar la intervención.</p>
<p>LOGRAR EL MEJORAMIENTO del problema, permitiendo además la transferencia y acumulación de los conocimientos aprendidos.</p>	<p>6. Desarrollar la intervención. 7. Evaluar los logros.</p>

Ilustración 2: Áreas y etapas de la resolución de problemas

Fuente: tomado de http://www.colombiaaprende.edu.co/html/home/1592/articles-189023_archivo_7.pdf

Según el estudio realizado por Jonassen en 1997 se presenta a la solución de problemas como una habilidad que incluye varios componentes cognitivos:

- ✓ Dominio del conocimiento (información proposicional, conceptos, reglas y principios).
- ✓ Estructura del conocimiento (redes de información, mapas semánticos, redes conceptuales, modelos mentales).
- ✓ Habilidades ampliativas (construcción y aplicación de argumentos, analogías, inferencias).
- ✓ Habilidades metacognitivas (establecimiento de metas, asignación de recursos cognitivos, evaluación del conocimiento previo y de la revisión del progreso/error).
- ✓ Componentes motivacionales/de actitud (esfuerzo ejercido, persistencia en la tarea, compromiso intencional).
- ✓ Conocimiento acerca de sí mismo (articulación del conocimiento previo, articulación del conocimiento sociocultural, articulación de las estrategias personales, articulación de los prejuicios/debilidades cognitivas).

(Date, 2018)

2.3.1. Proceso de pensamiento

Cada vez que se soluciona un problema es la acción mental en la cual el resultado representa una salida con algún grado de dificultad que alcanza un objetivo.

Gick en 1996 especifica tres conjuntos de procesos de pensamiento asociados a la solución de problemas:

- **El proceso de entendimiento:** implica la representación del problema, descomposición y su calificación
- **El proceso de búsqueda:** es la generación de alternativas de solución

- **El proceso de implantación:** es en donde los estudiantes prueban las soluciones que generaron.

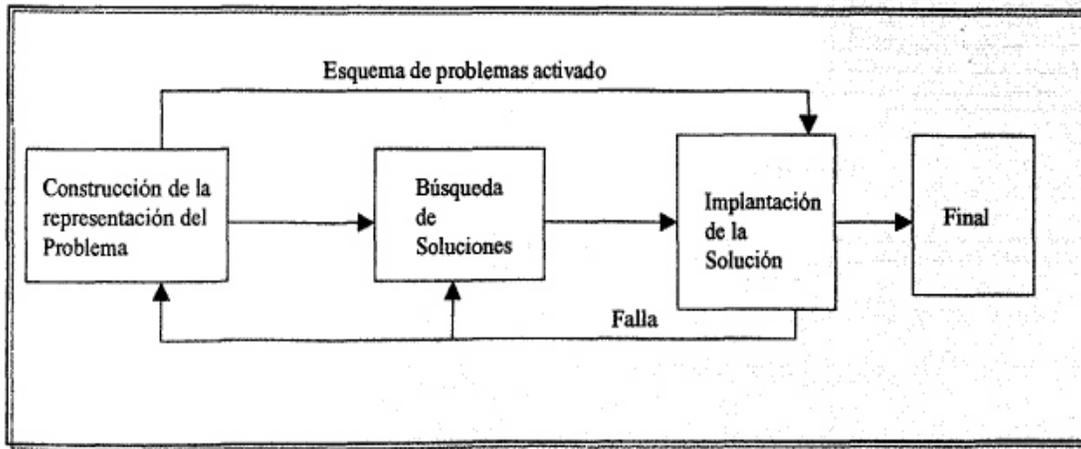


Ilustración 3: Esquema simplificado del Proceso de Solución de Problemas (Gick, 1986)

Fuente: tomado de <https://www.tandfonline.com/doi/abs/10.1080/00461520.1986.9653026>

2.4. Scratch

Scratch proviene de la palabra “Scratching” que significa “trozos de código que se pueden reutilizar fácilmente combinables y adaptados para nuevos usos” es un lenguaje de programación en bloques visual y modular desarrollado por Instituto Tecnológico de Massachusetts (MIT) en donde la primera versión salió en el año 2005 diseñado para que jóvenes y niños aprendan a programar de forma didáctica, profesores y académicos lo usan para enseñar a sus alumnos en las aulas de clase.

Su objetivo es que las personas que lo utilicen adquieran habilidades para la solución de problemas y realización de diseños, las habilidades desarrolladas incluyen:

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

(EDUTEKA, 2012)

Se puede instalar Scratch en ordenadores con sistemas operativos Windows, Ubuntu, Sugar, Mac o como una aplicación web desde el navegador preferido.

En la actualidad Scratch se ha perfeccionado, consiguiendo ser un lenguaje muy versátil y fácil de usar, esta herramienta es financiada por empresas muy importantes como Google, Microsoft, Intel, Dell, e incluso LEGO; ayudando a que tengan extensiones en donde se pueden interactuar con el manejo de robots creados con Lego Mindstorms, y con placas Arduino teniendo en cuenta que se pueden encontrar cursos y recursos que se ofrecen de manera gratuita.

(Juan Antonio Pascual, 2015)

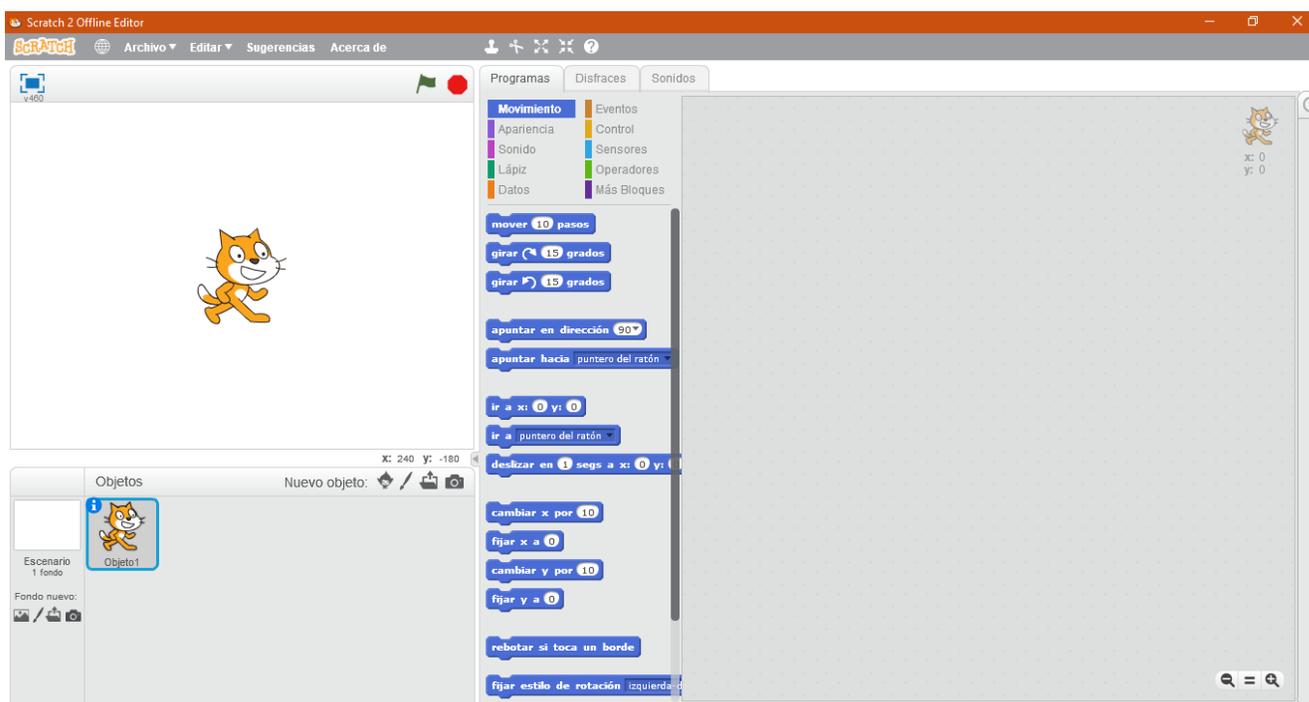


Ilustración 4: Entorno inicial de Scratch versión Offline

Fuente: imagen propia

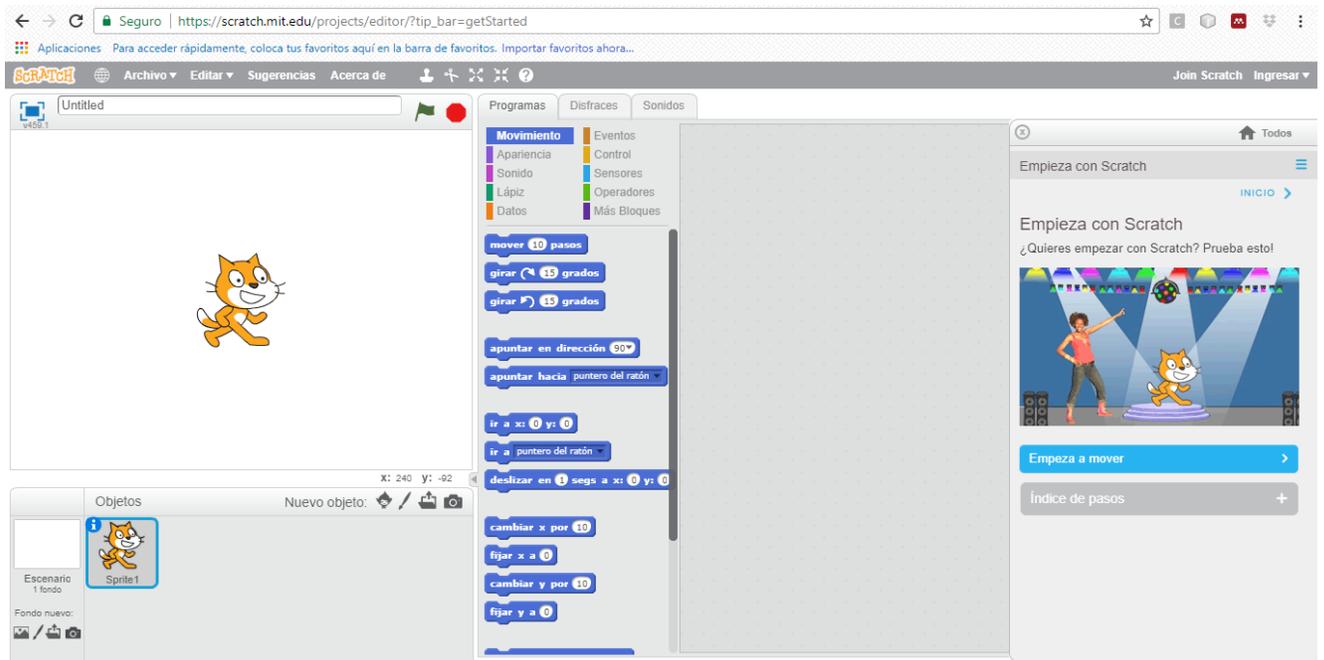


Ilustración 5: Entorno inicial de Scratch versión Online

Fuente: imagen propia

A la hora de realizar ejercicios de programación en donde se pueden crear una gran variedad de contenidos como presentaciones, historias interactivas, vídeos musicales, gráficas animadas, juegos, programas, etc., que pueden ser compartidos con otros usuarios. Para comenzar a trabajar con Scratch solo se deben tomar las piezas que contiene las ordenes o condiciones y encajarlas entre sí para que actúen de forma determinada.

Las ordenes o condiciones están divididas en categorías y son estas:

- **Movimiento:** Mover y girar un objeto por la pantalla.
- **Apariencia:** Cambiar la visualización del objeto: el fondo, hacerlo más grande o pequeño, etc.
- **Sonido:** Hacer sonar secuencias de audio.
- **Lápiz:** Dibujar controlando el tamaño del pincel el color y la sombra del mismo.
- **Datos:** Crear variables y su asignación en el programa.
- **Eventos:** Manejadores de eventos que inician determinadas acciones en un bloque.

- **Control:** Condicionales: Si no, por siempre, repetir y detener.
- **Sensores:** Los objetos pueden interaccionar con el entorno o con elementos creados por el usuario como un robot lego, por ejemplo.
- **Operadores:** operadores matemáticos, generadores aleatorios de números, cooperadores de posiciones.
- **Más bloques:** Bloques propios y controladores de aparatos externos.
(Resnick et al., 2009)



Ilustración 6: Categorías de instrucciones en Scratch

Fuente: imagen propia

También se puede encontrar una variante para niños entre 5 y 7 años no lectores o neolectores, llamado ScratchJr que es una app que se puede instalar en tablets, iOS y Android en donde solo basta arrastrar los bloques con el dedo, ideal para niños que no saben usar el ratón del ordenador, aunque en un ordenador se puede instalar gracias a un emulador de Android BlueStacks o con una aplicación ARC Welder de Google Chrome.

(Alberto Gómez & Chacón Larraz, 2016)

En esta variante nos podemos dar cuenta de la simplicidad que tiene para su uso en niños, una de estas cosas lo podemos encontrar es que desaparecen las coordenadas polares y aparecen una cuadrícula de 15 por 20 casillas en donde el objeto se puede arrastrar con el dedo



Ilustración 7: Entorno inicial de Scratch JR

Fuente: imagen propia

2.5. Institución educativa Bethlemitas Brighton

El 13 de abril de 1896 llegaron a Pamplona las primeras Bethlemitas y el 17 del mismo mes se instalaron y tomaron bajo su cuidado la Institución con el nombre de “ASILO DE LA SAGRADA FAMILIA, BRIGHTON”.

El terreno donde hasta el presente funciona fue obsequiado por él secretario episcopal Dr. Numa Julián Calderón a quien se debe la fundación del “Asilo Sagrada Familia Brighton”. a la Comunidad, que se trasladó a ella en 1906, cuando vino la R. M. Superiora General, María Luisa Salinas, recibiendo ella misma la escritura de la donación hecha por el benemérito Benefactor.

La institución empezó con 20 huérfanas, a las que se adiestraba en los oficios domésticos sin olvidar la instrucción religiosa e intelectual.

A las primeras Bethlemitas que trabajaron en esta obra, les tocó un período de mucha pobreza y privaciones, pero a pesar de las dificultades llevaron adelante la obra, hasta la llegada de la Madre Magdalena Restrepo, que lo transformó y le dio vida.

Con su exquisito don de gentes consiguió un buen número de benefactores que le prestaron valiosa ayuda.

El 17 de abril de ese mismo año se verificó la inauguración solemne de la institución bajo la presidencia del Ilustrísimo Señor Obispo diocesano y con asistencia del prelado de Mérida y de la mayor parte del clero de la ciudad.

Tomaron la palabra en ese acto el Ilustrísimo Señor Parra, el Doctor Numa Calderón, el venerable señor Cura del Carmen y el venerable Cura de Chinácota.

Años más tarde se cambió un poco la modalidad de la obra consiguiendo del gobierno la fundación de la Escuela Sagrada Familia en las mismas Instalaciones del Hogar con la básica Primaria, posteriormente fue creciendo hasta lograr la Básica Secundaria, como Centro Educativo; en el momento contamos con aprobación de la Secretaría de Educación como “Institución Educativa Bethlemitas Brighton” con orientación del SENA, en la Media Técnica

(10º y 11º). Cuenta con un número de 916 estudiantes, de Transición a Once grados y un excelente cuerpo de docentes, muy bien calificados y apto para el desempeño de su misión Bethlemitas.

(Huellas, 2010)



Ilustración 8: Entrada original Institución educativa Bethlemitas Brighton

Fuente: tomada de <http://mapio.net/pic/p-85869407/>

CAPITULO III

ESTADO DEL ARTE

3.1. Nivel internacional

A lo largo de todo el mundo son diversas las investigaciones que se han realizado en el campo del pensamiento computacional, aunque algunos investigadores difieren entre sí en los resultados obtenidos, es unánime la conclusión de que el aprendizaje de la programación de computadores es una de las herramientas más eficaces que se poseen en la actualidad para este propósito.

En diferentes países se observa cómo se vinculan a diferentes proyectos para el desarrollo del pensamiento computacional las universidades, gobiernos y algunas entidades privadas, quienes aportan sus recursos con el fin de estimular a los jóvenes al estudio de la programación y de esta forma desarrollar en ellos las habilidades propias del pensamiento computacional.

Se observa en España el proyecto titulado “El Pensamiento Computacional en la Enseñanza Obligatoria (Computhink)”, quien fue financiado por el Centro Común de Investigación (JRC) de la Comisión Europea y fue Lanzado en el mes de diciembre de 2015, este proyecto fue llevado a cabo por el Instituto de Tecnologías Educativas del Consejo Nacional Italiano de Investigación junto a European Schoolnet. Con el objetivo de contribuir al debate sobre la codificación, las habilidades transversales y las competencias a nivel europeo y de los Estados miembros, CompuThink está estrechamente vinculado con los estudios del Centro Común de Investigación sobre la competencia digital de los ciudadanos, los docentes y las organizaciones educativas.

Allí se propone que La introducción de los conceptos y habilidades fundamentales del Pensamiento Computacional en la enseñanza obligatoria requiere herramientas de aprendizaje que puedan hacer que las actividades de programación sean accesibles a los alumnos de primaria.

Donde se concluyó que existe un amplio consenso entre los expertos y los profesionales de que la introducción del Pensamiento Computacional en los currículos de todos los niveles educativos está originando la necesidad de un desarrollo profesional docente continuo.

Las actividades de formación a menudo están diseñadas específicamente para ser prácticas, con el fin de que los docentes puedan aplicar más fácilmente sus nuevas habilidades en sus clases.

También se logró concluir que para que la integración del Pensamiento Computacional en todos los niveles de enseñanza obligatoria sea completa, es necesario definir una visión clara y establecer metas específicas. Como el Pensamiento Computacional implica mucho más que ofrecer unas pocas horas de codificación, integrarla en el currículo requiere una estrategia sólida que tenga en cuenta la amplia gama de factores involucrados. Una consideración clave es la medida en que el Pensamiento Computacional se aplica a todo el espectro de áreas y asignaturas y, también, a contextos multidisciplinarios e interdisciplinarios. Es deseable que los conceptos del Pensamiento Computacional sean introducidos a los niños desde los primeros años de enseñanza primaria.

Estas consideraciones requieren un enfoque holístico de la integración del Pensamiento Computacional en la enseñanza obligatoria, que abarca aspectos esenciales tales como estrategias apropiadas de evaluación y una adecuada formación docente.

(Bocconi et al., 2016)

Los gobiernos también han notado la necesidad de compensar el desarrollo de las habilidades del pensamiento computacional desde tempranas edades en los colegios, tal es el caso de Inglaterra donde se ha implementado la enseñanza de la programación, desde los primeros años de estudio.

En el “National curriculum in England” se propone que una educación informática de alta calidad equipa a los alumnos para utilizar el pensamiento computacional y creatividad para comprender y cambiar el mundo. La informática tiene vínculos profundos con Matemáticas, ciencia, diseño y tecnología, y proporciona información sobre sistemas naturales y artificiales. El núcleo de la computación son las ciencias computacionales, en la que los alumnos aprenden los principios de información y computación, cómo funcionan los sistemas digitales y cómo poner este conocimiento en uso a través de la programación.

Basándose en este conocimiento y comprensión, los alumnos están equipados para usar la tecnología de la información para crear programas, sistemas y una gama de contenido. La informática también asegura que los alumnos se vuelvan digitales leer y escribir - capaz de usar, expresarse y desarrollar sus ideas a través de la información y tecnología de la comunicación, a un nivel adecuado para el futuro lugar de trabajo y como activos participantes en un mundo digital.

El plan de estudios nacional para la informática tiene como objetivo garantizar que todos los alumnos:

- puedan comprender y aplicar los principios y conceptos fundamentales de las ciencias computacionales, incluyendo abstracción, lógica, algoritmos y representación de datos.
- puedan analizar problemas en términos computacionales, y adquieren experiencia práctica en escribir programas de computadora para resolver tales problemas
- puedan evaluar y aplicar tecnologías de la información, incluidas las nuevas o desconocidas tecnologías, analíticamente para resolver problemas
- ser usuarios responsables, competentes, confiados y creativos de información y tecnologías de la comunicación.

Las metas plateadas para este currículo son grandes y se descomponen en dos etapas:

Primera etapa:

- entender qué son los algoritmos; cómo se implementan como programas en dispositivos digitales; y que los programas se ejecutan siguiendo instrucciones precisas y sin ambigüedades.
- crear y depurar programas simples.
- usar razonamiento lógico para predecir el comportamiento de programas simples
- utilizar la tecnología de forma intencionada para crear, organizar, almacenar, manipular y recuperar archivos digitales contenido.
- reconocer usos comunes de la tecnología de la información más allá de la escuela
- usar la tecnología de forma segura y respetuosa, manteniendo la privacidad de la información personal; identificar dónde buscar ayuda y apoyo cuando tienen dudas sobre el contenido o el contacto en Internet u otras tecnologías en línea.

Segunda etapa:

- Diseñar, escribir y depurar programas que logren objetivos específicos, incluido el control o simulando sistemas físicos; resolver problemas descomponiéndolos en pequeñas partes.
- usar secuencia, selección y repetición en programas; trabajar con variables y varios Formas de entrada y salida
- utilizar el razonamiento lógico para explicar cómo funcionan algunos algoritmos simples y detectar y corregir errores en algoritmos y programas
- comprender las redes informáticas, incluida Internet; cómo pueden proporcionar múltiples servicios, como la red mundial; y las oportunidades que ofrecen para comunicación y colaboración
- utilizar las tecnologías de búsqueda de manera efectiva, apreciar cómo se seleccionan y clasifican los resultados, y tener discernimiento en la evaluación del contenido digital

- seleccionar, usar y combinar una variedad de software (incluidos los servicios de Internet) en un rango de dispositivos digitales para diseñar y crear una gama de programas, sistemas y contenido que lograr los objetivos dados, incluida la recopilación, el análisis, la evaluación y la presentación de datos e información
- usar la tecnología de manera segura, respetuosa y responsable; reconocer comportamiento aceptable / inaceptable; Identificar una variedad de formas de informar inquietudes sobre contenido y contacto.

(Department for Education, 2013)

No solo en los diferentes países europeos se observa el interés en investigar y desarrollar proyectos sobre el pensamiento computacional, otro caso de este tipo de investigaciones se encuentra en Ecuador, donde la Universidad Politécnica Salesiana desarrollo en el año 2016 u proyecto titulado “Una aproximación desde la lógica de la educación al pensamiento computacional”.

Donde se plantea que la lógica de la educación aborda las políticas que en la materia se aplican en los sistemas educativos de las naciones y estudia los métodos empleados en la educación, en el aprendizaje del hombre, en las comunidades y en los resultados que se dan en las sociedades.

Esto ayuda a razonar sobre las falencias que se presentan en la educación y también en los buenos resultados que se consolidan en los sistemas educativos. Asimismo, constituye una propuesta académica en estos tiempos de cambio, innovación y transformación, que afectan a la sociedad hoy en día, para mejorar la acción pedagógica, el desarrollo y el progreso de la humanidad y de la sociedad con valores éticos y morales.

También se resalta en esta investigación que el pensamiento computacional: una competencia básica que permite abordar problemas para desenvolverse, en la sociedad actual digital, de manera inteligente y con imaginación, al combinar la abstracción y el pragmatismo, puesto que se basa en un mundo de ideas desarrolladas para un mundo real.

Hoy, los niños necesitan aprender y practicar nuevas habilidades, para tomar ventaja en la transformación revolucionaria, producida por los rápidos cambios tecnológicos que actualmente ocurren, y, de esta manera, en el futuro, hacer sus propias contribuciones con el fin de enfrentar los grandes retos del siglo XXI.

(Rincón Rueda & Ávila Díaz, 2016)

En la Universidad de Santiago de Chile donde se diseñaron once sesiones de trabajo, que permiten ir desde el nivel de reconocimiento y manejo de datos hasta automatización y simulación de problemas de la vida diaria, de complejidad media y alta. Las sesiones están graduadas jerárquicamente y en cada una de ellas se agregan habilidades que permitan ir alcanzando los niveles más altos del pensamiento computacional. Los resultados muestran que los recursos de actividades interactivos con Scratch, logran el objetivo planteado, motivando al usuario, y alcanzando al término de la experiencia niveles más altos del pensamiento computacional. Finalmente, la creatividad del estudiante prospera en las últimas sesiones, donde las actividades son más abiertas e interactivas que las iniciales.

En este trabajo se diseñó una estrategia para desarrollar el pensamiento computacional en personas, de los medios y formas revisadas, lo más recomendado fue por medio de juegos interactivos, pero más que jugar también creando estos juegos. La revisión de plataformas que permiten realizar juegos de forma lúdica nos llevó a Scratch, herramienta altamente utilizada hoy en día en los procesos de enseñanza interactiva, esta permite la generación de contenidos mediante una programación basada en bloques y en base a las funcionalidades que ofrece, se planifican los recursos y se desarrollan las actividades y se triangula con las componentes del pensamiento computacional. Los resultados obtenidos son alentadores y nos indican que estamos en el camino correcto, ya que se logra el objetivo propuesto y además las actividades interactivas que dejan una ventana abierta a la creatividad del aprendiz. El estudiante puede llegar a tener un pensamiento computacional avanzado con las actividades diseñadas, en forma lúdica y entretenida. Quisiéramos destacar la creatividad que tienen los estudiantes y como esta emerge en forma natural en las últimas sesiones, el concepto de hacer un juego para un estudiante es un desafío y las reglas que

ellos se imponen superan las expectativas de lo que se pretende. En este contexto, el juego una vez realizado es socializado y puesto a prueba por sus compañeros, resulta gratificante para el estudiante que su producto sea algo tangible y que otra persona lo quiera utilizar. También presenta un desafío su mejora para hacerlo lo más atractivo. En suma, las actividades interactivas para generar pensamiento computacional de alto nivel son una herramienta efectiva que ayuda a fortalecer los procesos lógicos que permiten la modelación correcta de solución de problemas, además de fomentar habilidades como la creatividad. Hay que seguir incursionando para establecer el efecto que provoca transversalmente en la resolución de problemas, en un ámbito más científico.

(Capot & Espinoza, 2015)

3.2. Nivel nacional

En Colombia el proceso de desarrollo del pensamiento computacional en diferentes instituciones a lo largo de todo el territorio nacional es un proceso que se ha desarrollado con lentitud, pero con paso firme, donde se observa de nuevo la intervención de las instituciones de educación superior en el desarrollo de este proceso, también se han sumado entes del gobierno nacional como el ministerio de educación nacional y el ministerio de las tecnologías de la comunicación y la información (TIC).

Un primer ejemplo de desarrollo de proyectos que buscan el desarrollo del pensamiento computacional en estudiantes de colegio es el que se desarrolla entre la Universidad del País Vasco UPV/EHU liderado por el Dr. Xabier Basogain, quien es el coordinador del proyecto, con la participación de la Red Nacional Académica de Tecnología Avanzada (RENATA), y el Ministerio de Tecnologías de la Información y las Comunicaciones (MinTIC), quienes han establecido un convenio para que este proyecto sea el primer pilotaje de una iniciativa que a corto plazo tenga un alcance a nivel nacional logrando que todos los estudiantes de los colegios de Colombia se beneficien de los aportes de esta investigación.

El objetivo del proyecto es permitir a los estudiantes y profesores que se familiaricen con los conceptos básicos del Pensamiento Computacional y a futuro poder incluirlos en el plan de estudios de las escuelas colombianas para lograr que las nuevas generacional no sean solamente consumidores de tecnología, sino productores de ella.

Este ambicioso proyecto de envergadura nacional ha dividido su público objetivo en cuatro grupos:

- Docentes, directivos de escuelas de primaria, secundaria, educación técnica y educación superior.
- Personal investigador en áreas afines a la computación, educación, y las ciencias cognitivas.
- Profesionales de la formación con interés en desarrollar contenidos formativos e-Learning para empresas o centros de enseñanza
- Responsables de la administración pública en materia de educación y tecnología.

El desarrollo del proyecto se desarrolla con la ayuda de la herramienta Scratch y de la plataforma Moodle, se busca que el método de aprendizaje sea mixto, es decir presencia del maestro en el aula de clase y apoyo las plataformas on-line, la fase inicial del curso se desarrolló en el 2016-2017 y una fase posterior en el 2017-2018.

En la Universidad Nacional de Colombia en el año 2016, se desarrolla un proyecto de Tesis para optar al título de Magister en Psicología, desarrollado por Luis Alfredo Sánchez Ruiz, donde se propone que. El pensamiento computacional es una habilidad esencial para la resolución de problemas en espacios diversos que van desde el uso de computadores hasta otros contextos de la cotidianidad. Para dar cuenta de esto, se han desarrollado múltiples programas educativos que involucran tareas de programación.

Para esta investigación se contó con estudiantes de dos colegios distritales de Bogotá que fueron el Colegio Distrital Manuel Cepeda Vargas y el Instituto Técnico Industrial Francisco José de Caldas. Los estudiantes del primer colegio se establecieron como grupo experimental, porque todos los sujetos de este colegio manejaron Scratch antes y durante

la aplicación. El segundo colegio se tomó como grupo control, en donde los estudiantes que participaron en la prueba no manejaban Scratch. Se debe aclarar que el término experimental se usa acá para facilitar la comprensión de los resultados pero que el diseño de este estudio es cuasi-experimental ya que los grupos no fueron asignados aleatoriamente a las condiciones.

Los resultados de este estudio sugieren que el uso de la herramienta de programación Scratch no solo fomenta el desarrollo de los conceptos computacionales descritos por Brennan y Resnick (2012) sino que también dejan ver que, la adquisición de dichos conceptos esta cernida a procesos de aprendizaje en los cuales incide tanto la formación continuada con el programa, como aspectos específicos de los conceptos de programación. De tal manera, por un lado, se destaca que los participantes del grupo experimental al estar involucrados en actividades de programación desde el curso octavo con Scratch, gradualmente adquirieron elementos importantes de pensamiento computacional y, por lo tanto, mostraron una mejor ejecución en los ejercicios planteados y a su vez en el manejo de los conceptos. Por el otro lado, los conceptos computacionales abordados se agrupan de tal manera que reflejan no solo grados de dificultad, sino aspectos de abstracción e integración de información que no se podrían apreciar en un usuario novato. Para comenzar, los resultados de las diferencias en los totales de la aplicación general, de la prueba de Scratch y la de LEGO por separado, indican que el desempeño de los estudiantes está significativamente asociado al uso de Scratch. En otras palabras, los conceptos computacionales que se plasmaron en los ejercicios de estas pruebas se han ido adquiriendo por el uso sostenido en el que se han involucrado los estudiantes del grupo experimental. Además, estas dos pruebas muestran cómo Scratch desarrolla las habilidades de pensamiento que fomentan la adquisición de los conceptos computacionales y también permiten evidenciar cómo se agrupan dichos conceptos. En este sentido se puede decir que esta agrupación está asociada a la dificultad que implican unos ejercicios sobre otros.

Sin embargo, también se puede afirmar que los conceptos de secuencias, evento, loops y paralelismos corresponden a una unidad conceptual que no se ha documentado, en donde estos primeros se dan por cadenas de acción de carácter lineal mientras que los conceptos de condicional, operadores y datos son nociones que implican estructuras con múltiples caminos.

No obstante, queda en evidencia que no hay diferencias significativas en la prueba de LEGO para los tres primeros ejercicios, lo cual indica que éstos pueden ser desarrollados sin tener un precedente específico con la interfaz de Scratch, que en cierta medida se podría explicar porque la solución para estos ejercicios en LEGO reside en la modificación del primer ejercicio cambiando uno o dos bloques al igual que en la prueba de Scratch. Aun así, en este caso las guías de correspondencias limitan la cantidad de opciones que pueden ser distractores en el caso de Scratch. Además de esto se puede decir que, en el caso de LEGO, es más sencillo para los estudiantes demostrar que son capaces de entender conceptos que van desde seguir una serie de pasos para que algo suceda (secuencias), pasando por identificar que hay acciones o situaciones que tienen causalidad sobre otras (eventos) y finalmente que hay formas de hacer que algo se repita las veces que se desee sin volver a hacer la programación desde el principio una y otra vez (loops o bucles). Lo cual podría indicar que este tipo de conceptos se pueden adquirir bajo otro tipo de herramientas o contextos cotidianos que no se indagaron en esta investigación. Mientras que, los conceptos de programación que implican un mayor nivel de abstracción y trabajo en la interfaz de Scratch (paralelismos, condicionales, operadores y datos), si reflejan un mejor desempeño por parte de los estudiantes del grupo experimental, dando a entender que aunque se evalúen estos conceptos con la prueba análoga en LEGO los resultados serán mejores para aquellos que lleven determinado tiempo involucrados con Scratch.

(Sánchez Ruiz, 2016)

Por su parte en la ciudad de Medellín la Universidad EAFIT en el año 2014, trabaja un Proyecto de investigación para optar el título de Maestría en Ingeniería con especialidad en tecnologías de información para educación, titulado “competencias mínimas en pensamiento computacional que debe tener un estudiante aspirante a la media técnica para mejorar su desempeño en la media técnica de las instituciones educativas de la alianza futuro digital Medellín”, quien fue desarrollado por Leidy Yoana Giraldo Gómez.

Donde se propone que todos los estudiantes al culminar la básica secundaria y continuar con su educación superior, deben desarrollar las competencias del siglo 21, permitiendo a los estudiantes colaborar, comunicarse, solucionar problemas y tener un pensamiento crítico; dentro de este tipo de habilidades se encuentra el Pensamiento Computacional.

La investigación buscaba establecer cuáles son las competencias en Pensamiento Computacional que requieren los estudiantes para ingresar a la media técnica en informática, con el fin de mejorar su desempeño, elaborando una propuesta de inclusión en el currículo de la educación básica primaria y básica secundaria, para las Instituciones Educativas de la Alianza Futuro Digital Medellín.

Allí el problema que se encontró fue que La media técnica en informática forma a los estudiantes en competencias específicas en diseño y desarrollo de software y para que este proceso se pueda llevar a feliz término, es necesario que los estudiantes aspirantes a la media técnica cumplan con el perfil de ingreso que se encuentra establecido por la Institución Educativa (Institución Educativa Félix de Bedout Moreno, 2014), además de tener las habilidades básicas en matemáticas, lógica y uso de herramientas ofimáticas; no obstante, se observa que los estudiantes que ingresan a la media técnica manifiestan algunas dificultades.

La investigación se desarrolla dentro de la modalidad de investigación descriptiva, en la categoría investigación proyectiva o proyecto factible. Se emplea este tipo de investigación, con el fin realizar un diagnóstico de la situación actual de las competencias en Pensamiento Computacional de los estudiantes de la media técnica y aspirantes a esta (grado noveno), un planteamiento de competencias y una propuesta de inclusión en el currículo de la básica

primaria y básica secundaria. Esta investigación se apoya en la investigación documental, con la revisión de materiales ya elaborados (Fuente secundaria), obteniéndose mediante una indagación bibliográfica con el fin de conocer el estado en que se encuentra la problemática, ubicar sus antecedentes y establecer las teorías en las que se fundamenta la investigación; además del estudio de campo, recolectando información directa de la realidad obtenida de la experiencia empírica y analizando e interpretando los resultados (Fuente primaria).

La presente investigación busca determinar las competencias en Pensamiento Computacional que deben tener los estudiantes de la media técnica para mejorar sus desempeños, especialmente en el módulo Desarrollo del pensamiento analítico sistémico, basado en la recopilación, análisis y depuración de la información para evaluar la viabilidad de la propuesta.

El público objetivo para este proyecto fue La Alianza Futuro Digital Medellín, está apoyada por el Ministerio de Educación Nacional, alcaldía de Medellín, la secretaria de educación municipal, Universidad Eafit, Intersoftware, CREAME, las Instituciones de Educación Superior articuladoras: Politécnico Colombiano Jaime Isaza Cadavid y el SENA y 20 Instituciones de educación media, buscando la "Transformación y articulación de la educación técnica-media y profesional y la tecnológica en el clúster del software" de la ciudad, en el marco de la política de transformación de la formación técnica y tecnológica del Ministerio de Educación Nacional.

La Alianza Futuro Digital Medellín, busca transformar los programas técnicos y tecnológicos de las Instituciones de educación superior al implementar la formación por competencias a través de ciclos propedéuticos para la educación media técnica, técnica profesional y tecnológica en áreas relacionadas con el desarrollo de software.

La investigación tiene como población objetivo algunas Instituciones Educativas que pertenecen a la Alianza Futuro Digital Medellín; sin embargo, se toma como muestra para aplicar los instrumentos y recolección de información algunas Instituciones Educativas que forman parte del convenio con el Politécnico Colombiano Jaime Isaza Cadavid, al tener

situaciones similares en cuanto a la dinámica de trabajo de la media técnica con la Institución Educativa Félix de Bedout Moreno, de donde surge la temática de investigación.

El análisis de los datos recolectados se analizó mediante se emplea el enfoque cuantitativo, a través de una comparación de resultados de las encuestas aplicadas a los dos actores: estudiantes de media técnica y estudiantes de grado noveno.

Al finalizar este proyecto se logró concluir que Se observa que el éxito del desarrollo de las competencias de Pensamiento Computacional en un país, región o institución depende en gran medida de la cultura organizativa o políticas Educativas que se implementan para la implantación en el currículo, además de las alianzas que se forman para garantizar la sostenibilidad en el tiempo.

El desarrollar en los estudiantes las competencias de Pensamiento Computacional permite que adquieran capacidades para resolver problemas, administrar información, trabajo en equipo, colaboración, creatividad, innovación, entre otras, consideradas también competencias o habilidades del siglo 21.

Desde el Ministerio de Educación Nacional no existe una directriz desde los estándares curriculares que establezca lineamientos para Pensamiento Computacional o Ciencias de la Computación para la escuela; No obstante, algunas de las áreas desarrollan algunos componentes de Pensamiento Computacional. Se considera que es muy importante para el país que el ministerio de educación nacional asuma su papel formulando políticas de estado para el desarrollo del Pensamiento Computacional en la educación básica.

(Yoana & Gómez, 2014)

El 29 de marzo de 2017, Mauricio Pérez Palencia quien es Licenciado en informática y medios audiovisuales por la Universidad de Córdoba, Colombia. Docente de educación básica y media. Maestrante en Educación. Sistema de Universidades Estatales del Caribe (SUE Caribe) (Colombia).

Público un documento titulado “el pensamiento computacional para potenciar el desarrollo de habilidades relacionadas con la resolución creativa de problemas”, escribe este

documento como un aporte científico a la didáctica de las tecnologías de la información y la comunicación, desde el cual se evidencian las potencialidades y privilegios pedagógicos que ofrecen las ciencias de la computación para el desarrollo de competencias relacionadas con la resolución creativa de problemas. Se lleva a cabo a través de un estudio cuasi experimental y desde una propuesta metodológica fundada en la programación de computadoras mediante el lenguaje de programación visual “Scratch” como principal herramienta pedagógica mediacional.

La investigación desarrollada por él se llevó a cabo en las instituciones educativas Rafael Núñez y Luis Villafañe Pareja, de los corregimientos San Andrés y Martín Alonso, del Municipio de Córdoba, departamento de Bolívar. La primera de ellas la posee un total de 570 estudiantes y la segunda 576, en los distintos niveles de preescolar, básica y media. Estas instituciones en los últimos años se perfilaron hacia la formación técnica, el caso del I.E Rafael Núñez forma desde el 2012, bachilleres con énfasis técnico en sistemas, en acompañamiento logístico y pedagógico con el Servicio Nacional de Aprendizaje SENA. Solo en 2015, sus egresados obtuvieron el título de Técnico en Sistemas emanado autónomamente por la institución, independientemente del acompañamiento del Sena. Situación similar atañe a Luis Villafañe Pareja, la cual forma con acompañamiento del Sena, bachilleres técnicos en Producción en procesamiento de frutas y hortalizas.

Al final de este proyecto se logró concluir que, partiendo de las condiciones estadísticas de los resultados, es posible determinar que la intervención pedagógica diseñada según parámetros metodológicos coherentes con el desarrollo del Pensamiento Computacional permitió el desarrollo habilidades del pensamiento relacionadas con la resolución creativa de problemas en estudiantes de básica secundaria.

La investigación fue un intento por contribuir al fomento de la computación cognitiva para la producción de nuevas innovaciones pedagógicas en contextos de educación escolarizada, promoviendo a manera de recomendación el ser pedagógicamente hábil para aprovechar las potencialidades de los avances en tecnología para el beneficio de experiencias educativas que optimicen la formación de los estudiantes dadas las exigencias de la actual era digital y la urgente sociedad del conocimiento. En este orden de ideas, al integrar la

programación de computadoras en los planes de estudio es posible garantizar el desarrollo de habilidades potenciadas y complementadas desde una formación apoyada en la codificación.

(Pérez, 2017)

En la ciudad de Medellín el proyecto titulado “Desarrollo del pensamiento computacional por medio de la creación de objetos virtuales de aprendizaje con Scratch por los estudiantes de media técnica para la educación básica primaria.” Que fue desarrollado por Alberto León Vásquez Giraldo, en la Institución Educativa Kennedy.

Este proyecto buscaba permitir a sus realizadores (estudiantes del grado 11º de Media Técnica en multimedia), durante un periodo de 2 meses, entender la importancia de lo que significa planear y llevar a cabo los pasos de una producción multimedia, lo importante que es la preproducción, la investigación del público objetivo al cual va dirigido un OVA, en este caso estudiantes de la básica primaria y el planteamiento de un objetivo de aprendizaje cuando se trata de contenidos educativos digitales para que suceda un verdadero aprendizaje significativo.

El proyecto se desarrolló en 7 etapas:

Etapa uno: Desarrollo del objetivo de aprendizaje, análisis y diseño de escenarios, bocetos personajes, historyboard, objetos gráficos, mecánicas del juego, contenidos y guion técnico a utilizar para la preproducción.

Etapa dos: Digitalización de personajes y escenarios en herramientas gráficas.

Etapa tres: Grabación y edición de los audios y diálogos necesarios para cada objeto utilizando micrófonos y software de audacity.

Etapa cuatro: Por medio de la programación en scratch los estudiantes de media técnica de grado 11º, crean objetos virtuales de aprendizaje en las diferentes temáticas de básica primaria seleccionadas dando prioridad al área de ciencias.

Etapa cinco: Los estudiantes de 11º aplicaron y validaron los objetos creados con dos grupos de una de las escuelas de primaria, ese mismo día se les enseñó a los niños a utilizar el programa de forma introductoria.

Etapa seis: Motivar a los niños a que continúen sus estudios en la programación con scratch. Luego de tener la experiencia de utilizar los OVAs, motivar a los estudiantes a aprender el manejo de esta herramienta y crear sus propios proyectos.

Etapa siete: publicar los OVAs en el sitio web del MIT para compartirlos con los compañeros y docentes como material de apoyo a sus clases y aprendizaje.

Al final del proyecto se concluyó que esta experiencia educativa se considera una práctica STEAM debido a que está pensada como un proyecto que contiene la intención no solo desarrollo del pensamiento computacional sino de las competencias STEAM de la siguiente forma: desde la ciencias al crear OVAs en sus temáticas lo que obliga a los estudiantes a investigar no solo en esta área sino los estándares educativos propuestos por el Ministerio de Educación Nacional para la básica primaria; desde la tecnología al utilizar software de diseño gráfico y de programación e integración con la plataforma arduino al crear josticks para los video juegos; desde la ingeniería al implementar la una metodología de desarrollo de software como lo es el Análisis – Diseño – Desarrollo – implementación de una aplicación informática; desde las artes al crear los bocetos de sus propios escenarios y personajes como una actividad que propicia el desarrollo de la creatividad e innovación y desde las matemáticas en el momento de la programación con scratch al utilizar la lógica matemática y el plano cartesiano en las animaciones y movimientos.

El pensamiento computacional es una tendencia muy joven que se dispone como una de las habilidades del siglo XXI que todas las personas deben tener para ser competitivo a escala mundial, que debe incluirse dentro del currículo y que se debe empezar a desarrollar desde una edad temprana, antes de llegar a la educación media técnica para el caso de la especialidad en desarrollo de software.

(León Vásquez, s/f)

3.3. Nivel Local

En el contexto local no se observan muchas investigaciones en el campo del desarrollo del pensamiento computacional, tan solo la Universidad de Pamplona a través del programa de Ingeniería de Sistemas se está comenzando a interesar por esta temática y posee un proyecto titulado “Laboratorio de Tecnologías de la Información (TI) para el desarrollo del pensamiento computacional como estrategia en la promoción de la ingeniería de sistemas en el Norte de Santander.”, este proyecto es liderado por el grupo de investigación del programa llamado “Ciencias Computacionales – CICOM”.

Este proyecto busca usar diferentes herramientas físicas y digitales con el fin de estimular, las habilidades del pensamiento computacional de un grupo de estudiantes que posean el perfil característico para ingresar al programa de Ingeniería de Sistemas de la Universidad de Pamplona, de esta forma no solo se busca encontrar potenciales talentos futuros, sino hacer promoción al programa en las diferentes instituciones de educación media de la región.

Como fase piloto de este proyecto se han dado algunos cursos cortos de Scratch a estudiantes de colegio, con apoyo de estudiantes del programa de Ingeniería de Sistemas que se encuentran en sus últimos semestres y deben realizar su trabajo social como requisito de grado, se ha observado interés por parte de algunas instituciones de educación media del municipio en participar en este proyecto y que sus estudiantes desarrollen sus habilidades del pensamiento computacional mediante el aprendizaje y uso de la programación en la herramienta Scratch.

(“Ingeniería de Sistemas - Unipamplona - Pensamiento Computacional”, s/f)

CAPITULO IV

METODOLOGÍA

4.1. Descripción de la metodología

En la Unidad 1, se realiza la justificación y descripción del problema que motivan el desarrollo de la presente investigación, así como lo objetivos de la misma. Se escribe a continuación el objetivo general para dejarlo en contexto “Determinar el impacto obtenido con el desarrollo de un curso de Scratch en el desarrollo del pensamiento computacional, en estudiantes del grado 11°, mediante la aplicación de pruebas técnicas y observaciones sobre campo”.

Para el desarrollo de la presente investigación se realiza inicialmente la selección de los participantes. En este caso se tiene en cuenta uno de los colegios de la ciudad que presenta mejores condiciones para el desarrollo de cada una de las actividades del proyecto (disponibilidad de instalaciones, tiempo, colaboración de la institución). El colegio seleccionado cuenta con dos grados 11. Con el objetivo de determinar si hay impacto en el desarrollo del pensamiento computacional gracias al curso de Scratch, se selecciona uno de los grupos para desarrollar el curso y el otro, como grupo de control.

Para encontrar el efecto obtenido luego de desarrollar el curso se realizan pruebas antes y después del curso a los dos grupos. También se hace uso de la observación directa para registrar la actividad observada a lo largo del desarrollo del proyecto cuando se presenta contacto con la muestra seleccionada, se registra la actividad con el fin de agrupar los diferentes perfiles de los estudiantes y comparar como estos afectan el desarrollo del pensamiento computacional.

El alcance será exploratorio y se llevó en un entorno real y natural de los estudiantes, interpretando cada uno de los fenómenos de cada persona involucrada en la investigación.

Los datos obtenidos para el análisis son variados, pero en esencia se encuentran en las narraciones del investigador del desarrollo de cada sesión, dichos datos se estructuran, agrupan y analizan estadísticamente con el fin de detectar patrones y observar aquellos factores que influyen en que un participante de la investigación obtenga resultados más favorables a otro que realiza el mismo proceso.

4.2. Población-muestra

Regularmente no se pretende generalizar los resultados obtenidos en la muestra a una población

Antes de seleccionar los estudiantes para la muestra de investigación se escogió entre las instituciones educativas el colegio Bethlemitas Brighton ya que este colegio, los curso que se tomó para la muestra son estudiantes de undécimo ya que se les puede medir de manera más puntual las habilidades de pensamiento computacional.

Los criterios para escoger esta institución fueron:

- La institución se tiene en media técnica en sistemas.
- Las clases de programación se realizarán en la institución educativa en las horas de técnica.
- La facilidad de los estudiantes para tener acceso a una computadora para realizar los trabajos.

La población escogida fueron estudiantes de grado undécimo entre las edades de 16 y 17 años, para la muestra se tomaron 25 estudiantes pertenecientes al Institución educativa Bethlemitas Brighton, esta institución se escogió porque tiene énfasis en sistemas.

Para la muestra se tomaron 25 de grado undécimo que es el grupo completo en donde se le aplicaron dos pruebas técnicas una al inicio y la otra al finalizar el curso.

El grupo de control se escogió el otro el otro once que son 24 estudiantes de la misma institución educativa escogido que al igual que la muestra se le aplicaron dos pruebas técnicas al inicio y al finalizar el curso.

Las clases se realizaron dentro del aula de clase de informática en donde los estudiantes realizaban ejercicios resolviendo unas guías de Scratch con ayuda de dos o tres tutores que les orientaban.

Semanal mente los estudiantes recibían una sección de dos horas, las guías estaban diseñadas para que se resolvieran completamente en las dos horas con ayuda de los tutores.

4.3. Herramientas utilizadas para la recolección de datos.

- Encuestas: se realizaron encuestas para determinar la predisposición de los estudiantes para el aprendizaje de programación en bloques.
- Pruebas técnicas: Se aplicaron dos pruebas técnicas, la inicial (previa al curso) y otra al finalizar (posterior al curso) a los grupos de control y al grupo experimental seleccionados con el fin de validar el grado de dificultad de la prueba.
- Diario de campo: se llevó un registro en cada sección en donde se llevan las fechas y la observación de las actitudes de los estudiantes describen la temática a desarrollar durante cada sección en medio digital las (anotaciones de la observación directa).
- Fotografías: también se tomaron como evidencias fotografías de los estudiantes trabajando.

4.4. Recolección de datos

- Participación pasiva: la participación en este caso del observador fue pasiva ya que la interacción del investigador y el grupo fue nula para evitar el enfoque como observador.
- Muestras: como muestras se realizaron encuestas a todos los estudiantes de los dos grupos (control y experimental)
- Sesiones: se realizaron siete secciones con el grupo de 11B, con 2 horas duración cada y una sección diaria a la semana, el lugar establecido fue el aula e informática de la institución elegida.
- Agenda: se llevó una agenda especificando la hora, fecha y lugar de las secciones realizadas.
- Documentos: como documentos se tienen fotos que se tomaron en cada sesión aparte de las fotos se adquirió proyectos en Scratch que realizaron al finalizar el curso.
- Pruebas técnicas: se elaboraron dos pruebas técnicas que fueron revisadas por uno de los profesores de la universidad de Pamplona, estas pruebas se realizaron.
- Encuestas: son de tipo pre-estructurado, esta encuesta pretende determinar qué porcentaje de los estudiantes presentaban características específicas para aprender programación

Las características eran facilidades con la lógica y las matemáticas también se quería determinar la cantidad de estudiantes les podía interesar el curso según sus presencias para estudiar carreras profesionales.

4.5. El análisis de los datos

- Diario de campo: los análisis de los datos se determinaron:
 - por la cantidad de preguntas durante las sesiones de los estudiantes
 - El interés que demostraron desde el inicio hasta el final de las secciones.
 - Interés por profundizar en el tema.
 - Cantidad de estudiantes con apatía a las secciones.

- Comportamientos recurrentes de los estudiantes durante las siete sesiones.
- Pruebas técnicas: la prueba técnica se realizó en formulario Google y para analizar los datos fue por medio de Excel en donde según la respuesta se contaban cuantos alumnos contestaron la correcta y cuantos la incorrecta en las dos pruebas.
- Fotografías: se organizaron por tipo de datos
- Encuestas: Como las encuestas como estas se realizaron en formularios Google se descargaron y para analizarlas dependiendo de la pregunta se separó:
 - Según respuestas similares se dividió en categorías.
 - Tipo de pregunta si era una pregunta con opciones o si la pregunta era abierta.
 - Concurrencia en palabras cuantos alumnos contestaron de manera similar a una determinada pregunta

CAPITULO V

DESARROLLO DE LAS ACTIVIDADES

Como estrategia para la divulgación del material de trabajo, pruebas técnicas y la encuesta de predisposición para el aprendizaje de la programación de computadores, se creó un blog el cual cumple únicamente el papel de servir de intermediario entre el material propuesto para el desarrollo de cada sesión y los tutores que dirigieron las sesiones.

La dirección del blog es <http://pensamiento-comp-2018.blogspot.com.co/>

5.1. Diseñar encuesta de predisposición para el aprendizaje de la programación de computadores

La encuesta que se diseñó buscaba, encontrar afinidades entre las experiencias de los participantes del curso con las tecnologías de la información, su facilidad en las habilidades lógico-matemáticas, básicas para posteriormente confrontar estos resultados con el desempeño a lo largo del curso.

Para la realización de la encuesta se utilizó el servicio de formularios en línea de Google, la encuesta aplicada consta de 7 preguntas distribuidas de la siguiente manera:

- 2 preguntas de casillas de verificación.
- 4 preguntas de selección única
- 1 pregunta abierta.

El enlace en el cual se creó la encuesta es el siguiente:
https://docs.google.com/forms/d/17_WK5KNs9nHbZUp4HRbQ2-d6cVRHIBRCn2tMPNq51jl/edit

A continuación, se muestran las preguntas que se realizaron en la encuesta:

Encuesta de predisposición para el aprendizaje de la programación

Nombre: _____

Grado: _____

La siguiente encuesta está dirigida a estudiantes de educación media del municipio de Pamplona con el fin de determinar que predisposición poseen estos para el aprendizaje de la programación, por favor lea atentamente cada una de las siguientes preguntas y responda de la manera más honesta posible, la información aquí obtenida será totalmente confidencial y solo se trabajara con los datos con fines de investigación.

1. Marque las casillas de las tecnologías con las cuales usted ha interactuado.
 Smartphone Tablet Computador
 Facebook Instagram G + Twiter Snapchat
 Video Juegos (Play Station, XBox, etc.) Robótica
 Gadgets (Robots, Drones, Carros Control remoto, Etc.)
2. ¿Le causa curiosidad la forma en que fueron creadas estas tecnologías?
 Sí No
3. ¿Le gustaría aprender a desarrollar las aplicaciones utilizadas en dichas tecnologías?
 Sí No
4. Si la respuesta anterior fue "SI" cual o cuales son las tecnologías que le gustaría aprender a crear.
 Aplicaciones para Smartphone Aplicaciones para Tablet
 Aplicaciones para Computador Redes Sociales.
 Video Juegos Creación de Gadgets Robótica
5. ¿Tiene facilidades o interés en la Matemática y/o la lógica matemática?
 Sí No
6. ¿Dentro de sus planes futuros que tipo de carrera le gustaría estudiar?
 Ingeniería Economía Salud Ciencias Políticas Artes y Humanidades Otra Cual: _____
7. ¿Cuál cree que es el perfil laboral de un ingeniero de sistema?

5.2. Aplicar la encuesta a los estudiantes de colegio

La encuesta se aplicó en las instalaciones del colegio en el espacio concedido por este para la aplicación de la misma, la prueba fue aplicada tanto al grupo de control como al grupo experimental en las mismas condiciones, pero por separado.

5.3. Tabular resultados de la encuesta

Los resultados obtenidos con la encuesta se tabularon con ayuda de la herramienta para formularios de Google, a continuación, se presentan con ayuda de gráficos los resultados de los mismos.

1. Marque las casillas de las tecnologías con las cuales usted ha interactuado.

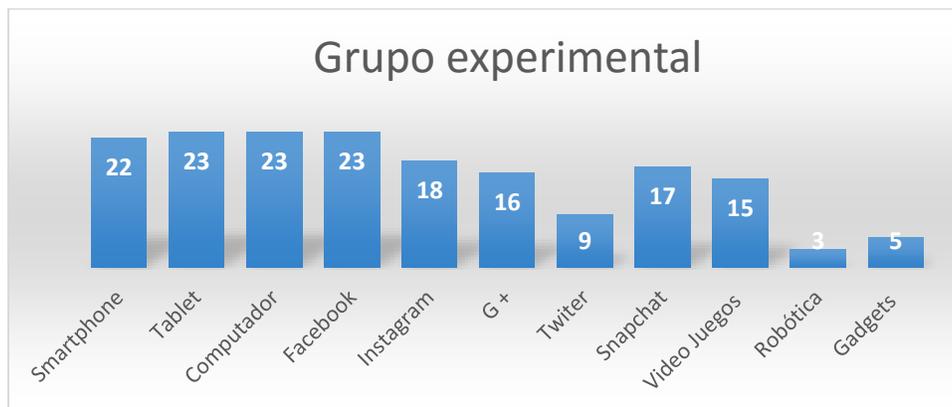


Ilustración 9: Tecnologías usadas grupo experimental

Fuente propia



Ilustración 10: Tecnologías usadas grupo control

Fuente propia

2. ¿Le causa curiosidad la forma en que fueron creadas estas tecnologías?

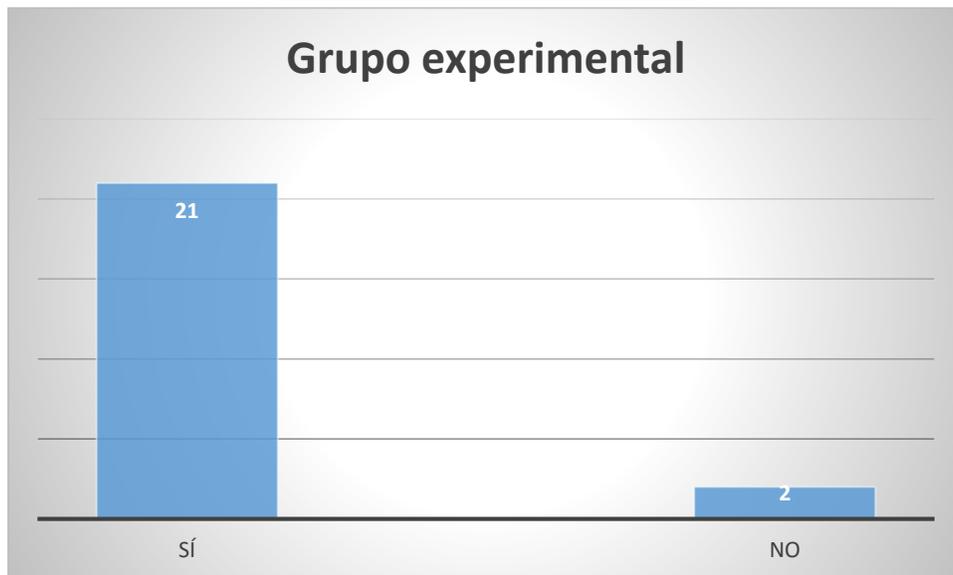


Ilustración 11: Curiosidad por la creación de tecnologías experimental

Fuente propia

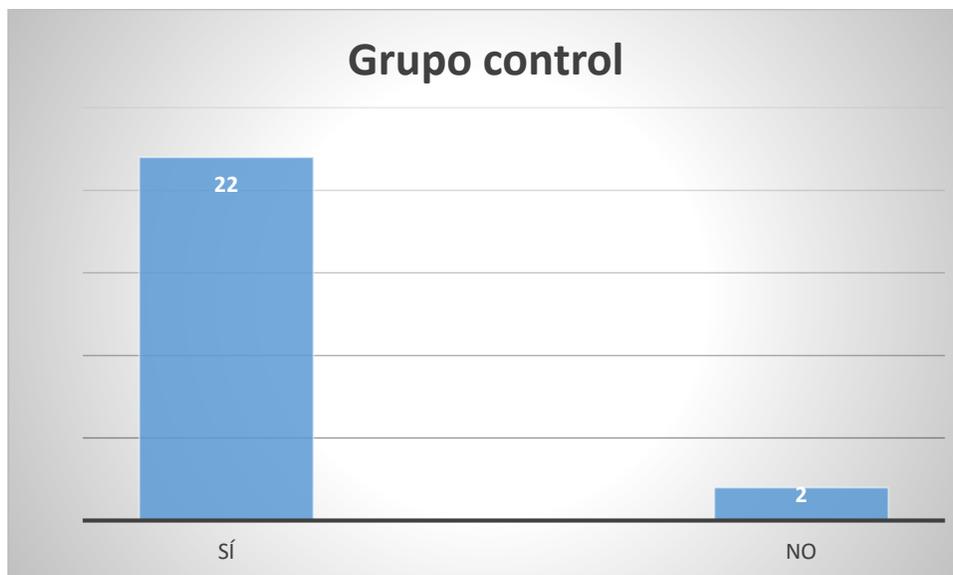


Ilustración 12: Curiosidad por la creación de tecnologías grupo control

Fuente propia

3. ¿Le gustaría aprender a desarrollar las aplicaciones utilizadas en dichas tecnologías?

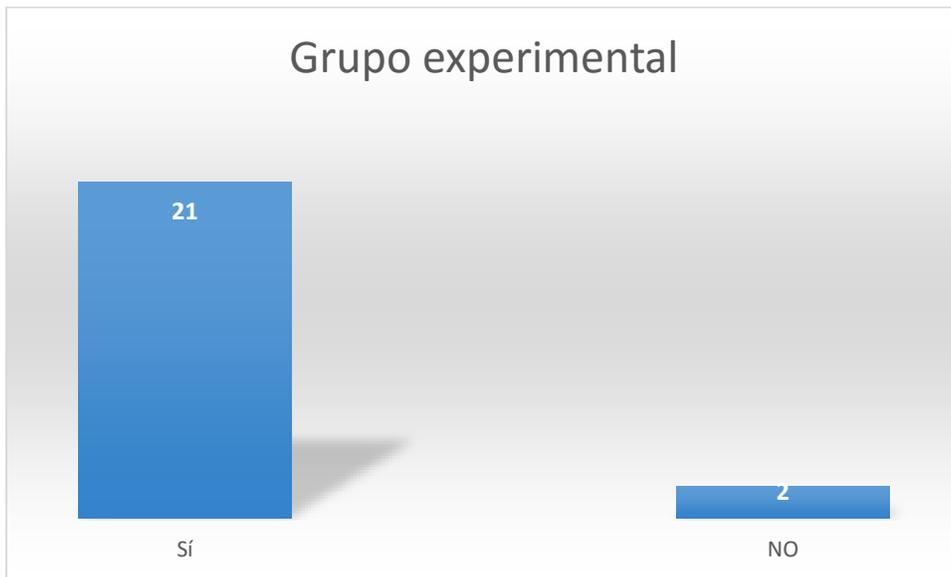


Ilustración 13: Interés por aprender a desarrollar nuevas tecnologías grupo experimental

Fuente propia

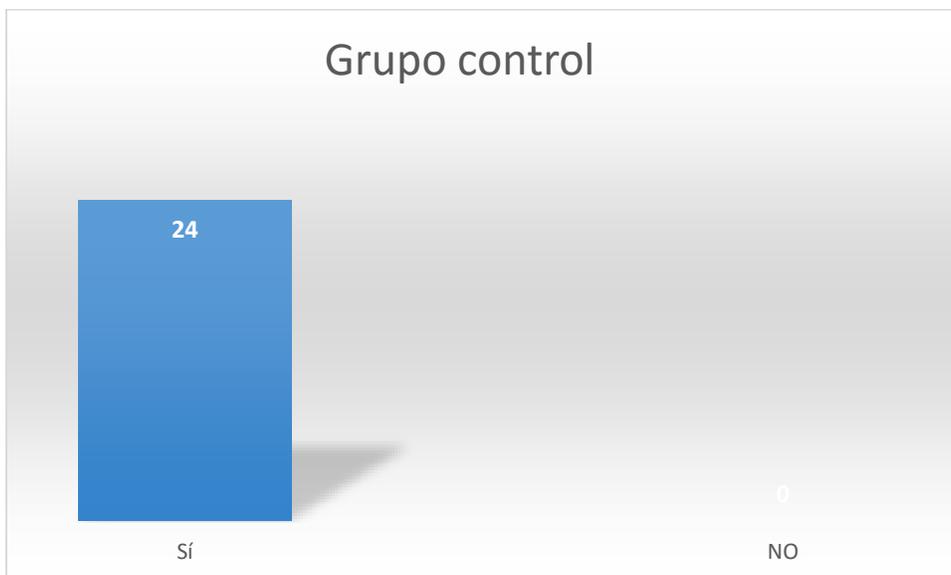


Ilustración 14: Interés por aprender a desarrollar tecnologías grupo de control

Fuente propia

4. Si la respuesta anterior fue "SI" cual o cuales son las tecnologías que le gustaría aprender a crear.

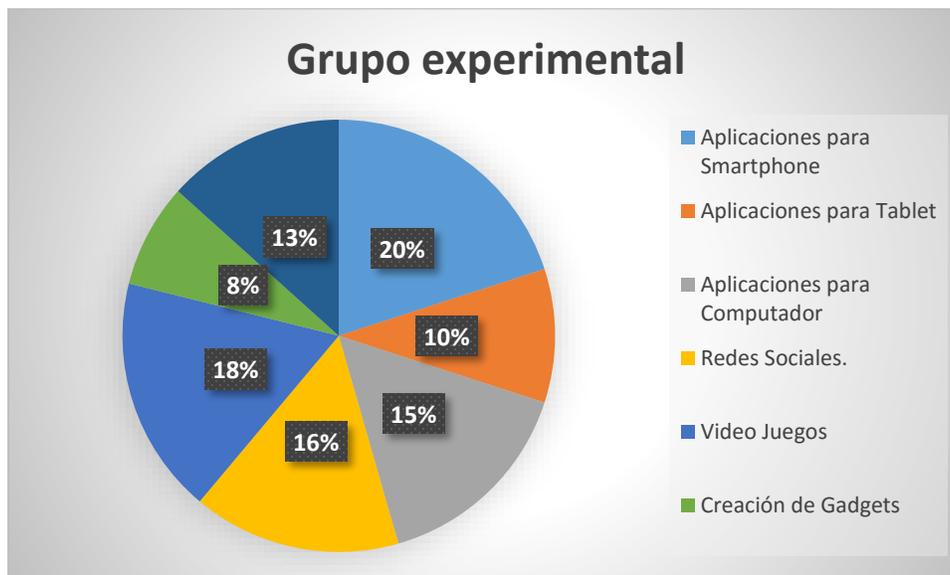


Ilustración 15: Tecnologías que le gustaría aprender grupo experimental

Fuente propia

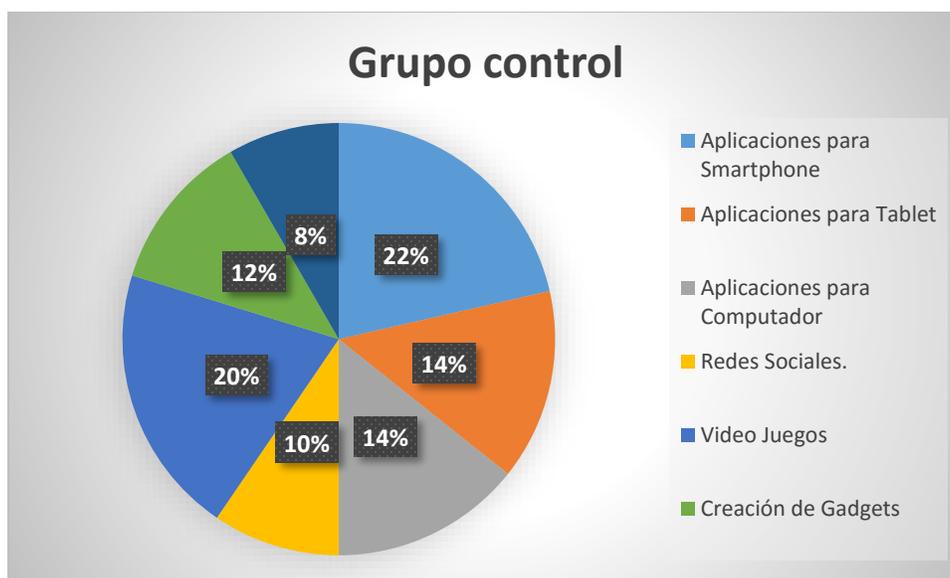


Ilustración 16: Tecnologías que le gustaría aprender grupo de control

Fuente propia

5. ¿Tiene facilidades o interés en la Matemática y/o la lógica matemática?



Ilustración 17:Facilidad o interés por matemáticas y/o lógica matemáticas

Fuente propia



Ilustración 18:Facilidad o interés por matemáticas y/o lógica matemáticas

Fuente propia

6. ¿Dentro de sus planes futuros que tipo de carrera le gustaría estudiar?



Ilustración 19: Carreras a estudiar grupo experimental

Fuente propia



Ilustración 20: carreras a estudiar grupos de control

Fuente propia

7. ¿Cuál cree que es el perfil laboral de un ingeniero de sistema?



Ilustración 21: Perfil de ingeniero de sistemas según grupo experimental

Fuente propia



Ilustración 22: Perfil de ingeniero de sistemas según grupo

Fuente propia

5.4. Diseñar la prueba técnica inicial y la prueba técnica final para medir el pensamiento computacional

Para el diseño de la prueba se tomó como base las habilidades del pensamiento computacional descritas anteriormente, y sobre cada una de ellas se formularon dos preguntas para cada prueba técnica.

- **Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva:** para el análisis de esta habilidad se desarrolló una serie de preguntas de decisión, que buscan observar como el estudiante razonaba y afrontaba la toma de decisiones en diversas circunstancias.

Prueba técnica inicial:

1. Un joven desea ir del extremo inferior izquierdo (punto A) de un parque a su extremo superior derecho (punto B), se sabe que todos los lados del parque tienen la misma longitud, ¿cuál es el camino más corto para que el joven llegue desde el punto A hasta el punto B, y que porcentaje de camino se ahorra este al ir por el camino más corto?

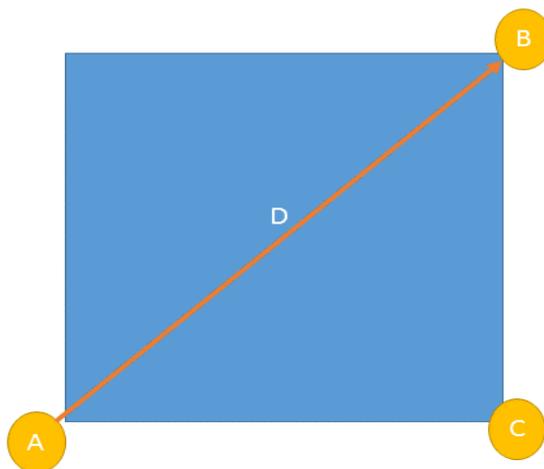


Ilustración 23:Imagen de referencia pregunta 1 técnica inicial. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva

Fuente propia

- a) Ir de A hasta C y luego de C hasta B con un 10% de ahorro dado que todos los lados son iguales, todos los caminos tendrán la misma longitud.
- b) Ir de A ha B pasando por la ruta D (línea roja), con un 50% de ahorro ya que al pasar por una diagonal y ser un cuadrado perfecto esta mide lo mismo que uno de sus lados.
- c) Ir de A ha B pasando por la ruta D (línea roja), con un ahorro menor de 1% ya que no sin importar la ruta las distancias no cambian significativamente.
- d) Sin importar que ruta se tome siempre tendrá la misma longitud por lo que el ahorro será de 0% en todas las rutas.

2. El siguiente diagrama describe la distancia y el costo de viajar por algunas ciudades de Colombia, diga cuál es el camino más económico para ir de Pamplona hasta Cali.

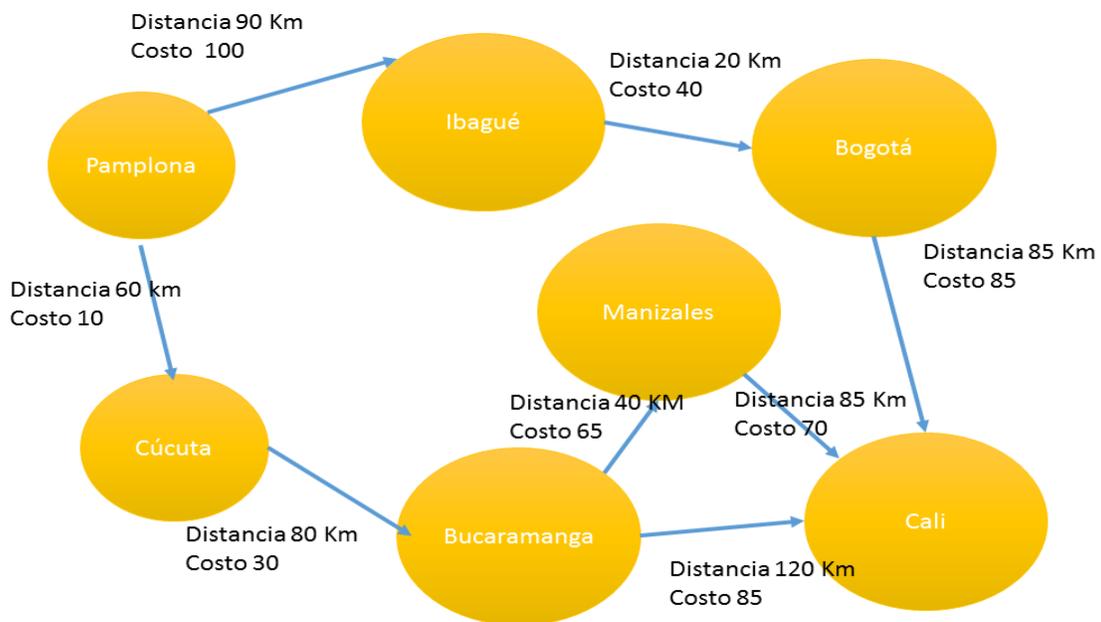


Ilustración 24: Imagen de referencia pregunta 2 prueba técnica inicial. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva

Fuente propia

- a) Pamplona, Ibagué, Bogotá y Cali.
- b) Pamplona, Cúcuta, Bucaramanga, Manizales, Cali.
- c) Pamplona, Cúcuta, Bucaramanga, Cali.
- d) Pamplona, Cúcuta, Manizales, Cali.

Prueba técnica final:

- 1) Para ir del punto A al punto B se debe escoger una ruta de que pasa por las ciudades C, D, E, F, G, H ¿cuál de las rutas que se observa en el diagrama es la más corta en cuanto a distancia?

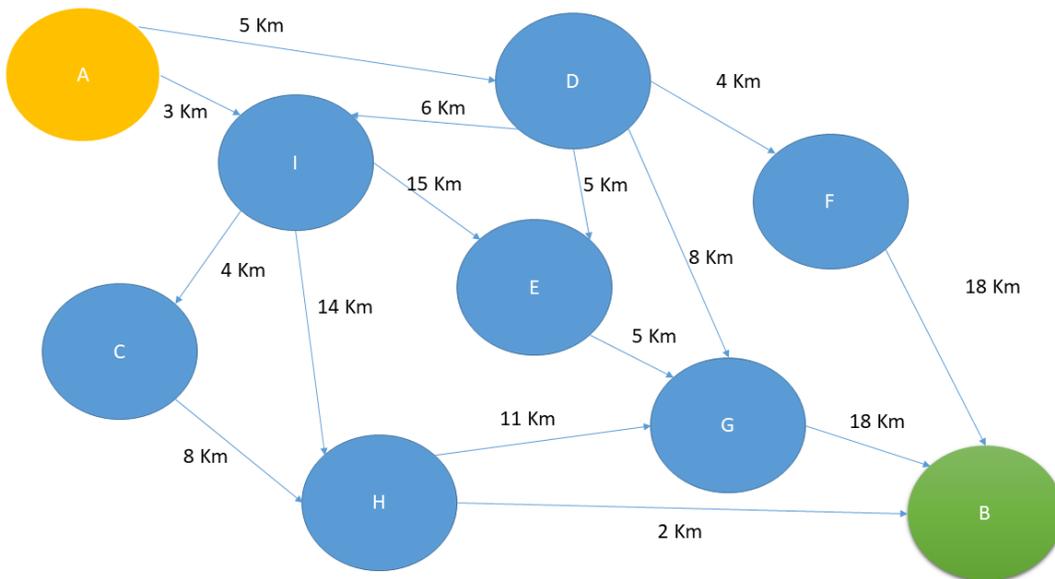


Ilustración 25 imagen de referencia prueba técnica final. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva

Fuente propia

- a) A, I, E, G, B
- b) A, D, F, B
- c) A, I, H, B
- d) A, I, C, H

2) Usted desea viajar a Madrid (España) de regalo de grado, para ello se le ofrecen varias opciones para realizar este viaje, debió a que Pamplona (Colombia) se encuentra muy lejos de Madrid se deben tomar varios medios de transporte en diferentes distancias, a continuación, se presenta la información de rutas, precios y tiempos suministrados por la agencia de viajes para realizar dicho desplazamiento.

# ruta	Ruta	Medio de transporte	Precio	Duración de trayecto
1	Pamplona-Cúcuta	Bus	\$10.000	2 horas
2	Pamplona-Cúcuta	Taxi	\$15.000	1.5 horas
3	Pamplona-Cúcuta	Helicóptero	\$1.000.000	20 minutos
4	Cúcuta- Bogotá	Bus	\$120.000	12 horas
5	Cúcuta- Bogotá	Taxi	\$200.000	10 horas
6	Cúcuta- Bogotá	Avión	\$350.000	2 horas
7	Bogotá-Madrid	Avión	\$2.000.000	12 horas
8	Bogotá-Madrid	Caminando	\$0	2 meses

Tabla 2: pregunta 2 prueba técnica final. Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva

En base a la información anterior cuales son las rutas a seguir teniendo en cuenta que se prioriza el ahorro de tiempo sobre el costo del desplazamiento.

- a) 1,4,8
- b) 1,6,7
- c) 3,5,6
- d) 3,6,7

- **Organizar datos de manera lógica y analizarlos:** Esta habilidad fue evaluada con preguntas que buscaban que dada cierta información el estudiante la clasificara y utilizara de manera adecuada, según cada pregunta.

Prueba técnica inicial:

1. Diga cuál es el orden correcto de la siguiente información.

La papelería de su barrio debe declarar sus ventas anuales ante la Dian con propósito del cobro de impuestos, el encargado de llevar la contabilidad de la papelería confundió los datos de las ventas con los de años anteriores, para efectos de simplificar la contabilidad el encargado identificaba 4 grupos de productos ofrecidos por la tienda, así como un valor promedio para cada grupo

Producto	Precio(del año que se va a declarar)
Libros(textos escolares)	\$50.000
Lápices (Lapiceros, colores, lápices, etc.)	\$3.000
Cuadernos (diferentes tipos)	\$5.000
Morrales (diferentes tipos)	\$65.000

Tabla 3: pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Se sabe que el tipo de productos más vendido (Por unidades) fue los lápices, pero los morrales brindaron la mayor cantidad de dinero en ventas, por su parte los libros generaron la menor cantidad de ventas que fue superada ampliamente por las ventas de cuadernos y fue solo un poco menor que lo vendido en lápices. ¿Cuál el orden de los productos según la cantidad de dinero que generaron?

a)

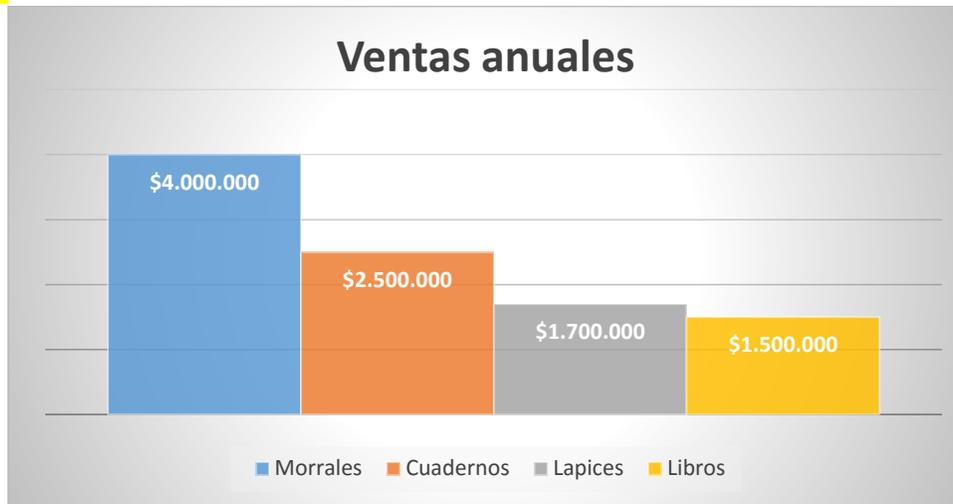


Ilustración 26: Opción de respuesta a pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Fuente propia

b)

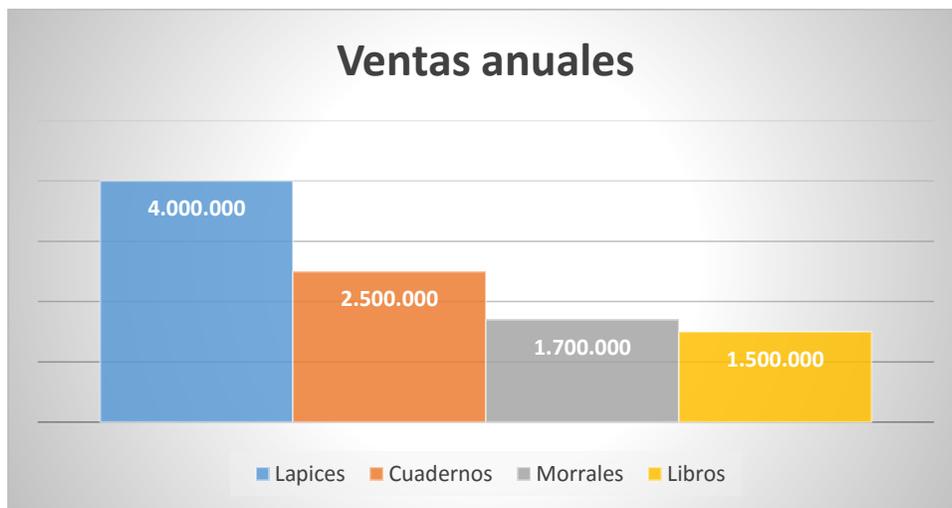


Ilustración 27: Opción de respuesta b pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Fuente propia

c)

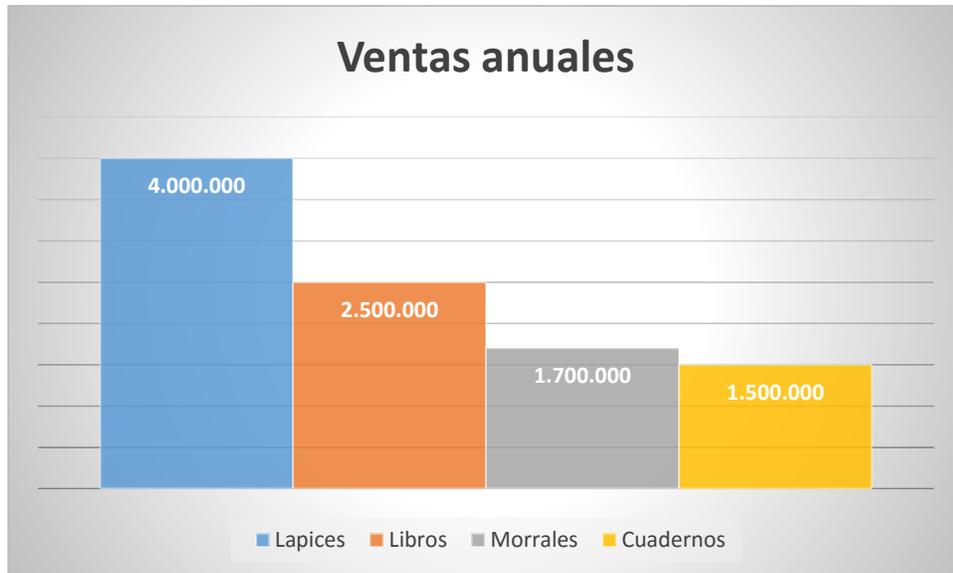


Ilustración 28: Opción de respuesta c pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Fuente propia

d)



Ilustración 29: Opción de respuesta d pregunta 1 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Fuente propia

2. Se afirma que el recipiente 1 se llena con 3 veces la cantidad del recipiente 3, mientras que el recipiente 4 posee la mitad de la capacidad de almacenamiento del recipiente 3, por su parte el recipiente uno puede ser llenado con el recipiente 4 con tan solo 6 veces el volumen de este, si se sabe que la capacidad del recipiente 2 es de 3 litros, ¿cuál es la capacidad de los demás recipientes?

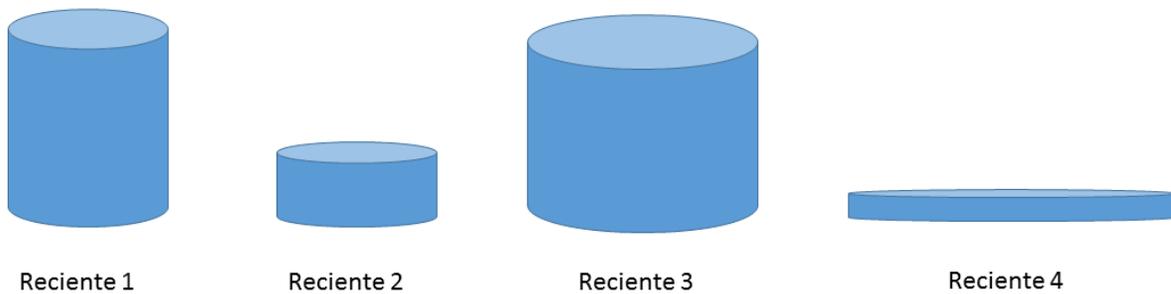


Ilustración 30: imagen e ilustración pregunta 2 prueba técnica inicial. Organizar datos de manera lógica y analizarlos

Fuente propia

Nota: dado que el material de los 3 recipientes es diferente, el volumen que cada uno de estos puede acumular puede o no estar relacionado con su tamaño.

- a) 3, 3, 1, 0.5
- b) 6, 3, 1.5, 1
- c) 2, 3, 6, 7
- d) 1, 3, 0.5, 2

Prueba técnica final:

1. Durante una excursión usted se pierde en el bosque y se encuentra completamente solo, dado que antes de dicha excursión usted no había informado a nadie de sus planes, no tiene esperanzas de ser rescatado dado que nadie conoce su ubicación.

Pero usted es un experto en supervivencia y posee la siguiente información sobre el bosque donde se encuentra perdido:

Temperatura: Muy caliente

Humedad: Alta

Tipo de vegetación: mayormente arboles grandes y frondosos, variedad de frutas silvestres.

Tipo de fauna: Gran variedad de animales aptos para consumo humano, Alta tasa de encuentro con animales peligrosos.

Disponibilidad de agua: baja, el rio más cercano está a 3 días de camino, y lo aleja 2 días más de la civilización.

Lejanía de la civilización: el pueblo más cercano está a 7 días de camino.

Luego de revisar sus anotaciones usted decide realizar una lista en la cual establece sus prioridades y como las puede suplir con sus habilidades

Prioridades	Habilidades
Comida	Recolección de frutas y pesca con mosca
Refugio	Construcción de choza en el suelo (toma 1 día) Construcción de choza en un árbol(toma 3 días)
Caminar	En condiciones óptimas (5 km diarios)
Prender fuego	Le toma (1/2 día)
Agua	Recoger el agua de lluvia
Orientación	Experto en manejo de la brújula

Tabla 4: pregunta 1 prueba técnica final. Organizar datos de manera lógica y analizarlos

Por ultimo usted realiza una lista de todas las cosas que puede hacer para definir un itinerario a seguir.

1 Construir choza que le tome 1 día

2 Construir choza que le tome 3 días

3 Prender el fuego

4 capturar un animal con arco y flecha

5 caminar hasta el rio

6 recoger frutas

7 pescar con mosca

8 caminar hasta la civilización

Dado que usted posee agua para 4 días de viaje y usted se perdió en la mañana, ¿cuál es el itinerario a seguir para lograr sobrevivir?

a) 2,5,3,7,8

b) 1,4,6,5,8

c) 3,1,6,5,8

d) 2,5,6,7,8

2. A continuación, se muestra el diagrama de flujo de uno de los sistemas de seguridad de un nuevo prototipo de auto, el cual posee la característica de manejarse solo, el esquema mostrado es solo un bosquejo básico de cómo se desea que el auto maneje los cambios de velocidad.

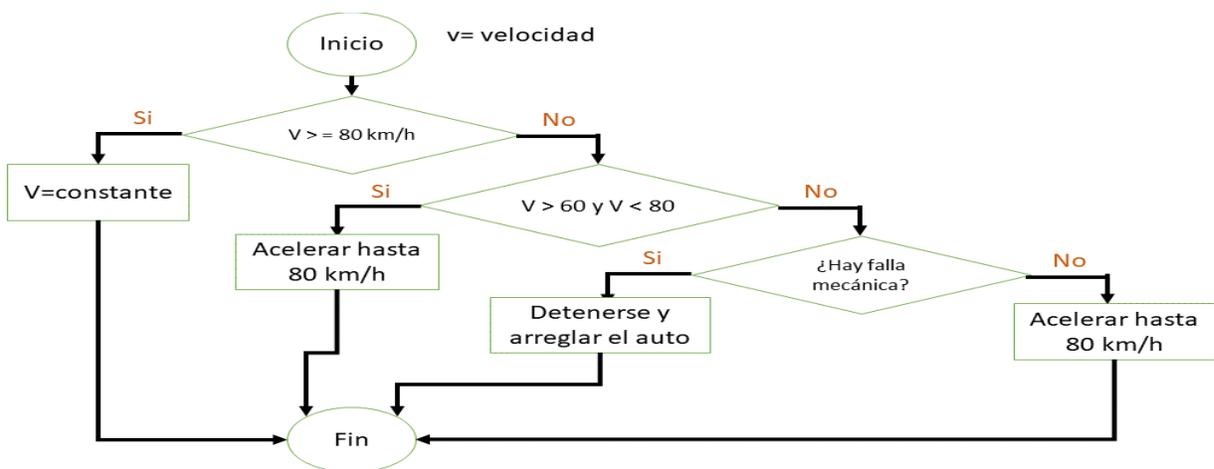


Ilustración 31: Pregunta 2 prueba técnica final. Organizar datos de manera lógica y analizarlos

Fuente propia

Según el diagrama ¿a qué velocidad debe viajar un vehículo para que a este se le realice una revisión mecánica?

- a) Mayor o igual a 80 km/h
- b) Entre 60 km/h y 80 km/h
- c) Menor de 60 km/h
- d) Nunca se ejecutará la acción de la revisión mecánica

- **Automatizar soluciones mediante pensamiento algorítmico:** Se utilizaron preguntas enfocadas al razonamiento numérico, que brindaron información sobre las habilidades algorítmicas de los estudiantes.

Prueba técnica inicial:

1. Un lechero debe medir exactamente 1 litro de leche, pero solo cuenta con una cantina de 3 y otra de 5 litros, el lechero no puede desperdiciar ni una sola gota de leche y la medición deberá ser lo más exacta posible. ¿Cuáles pasos debe seguir el lechero para poder medir el litro exactamente?

- 1. Llenar la cantina 3 litros.
- 2. Llenar la cantina de 5 litros.
- 3. Agregar la leche de la cantina de 3 litros a la de 5 litros.
- 4. Agregar leche de la cantina de 5 litros a la de 3 litros.
- 5. Tener 1 litro exacto.

Tenga en cuenta que los pasos descritos anteriormente no corresponden a un orden preestablecido, los pasos se pueden repetir y pueden estar o no todos incluidos en la solución del problema, seleccione cuál de las siguientes secuencias de pasos llevan al lechero a medir el litro exactamente.

a) 1,3,1,3,5

b) 1,2,3,4,5

c) 2,3,1,2,5

d) 2,4,2,3,5

2. Para navidad usted realizo un largo viaje a un destino turístico del país, el viaje se inicia a una velocidad de 45 kilómetros por hora y se mantiene así durante 2 horas, luego de esto se llega a una parte de la carretera que se encontraba en óptimas condiciones por lo que la velocidad del vehículo aumento a 80 kilómetros por hora y se mantuvo con esta velocidad constante durante el doble de tiempo que el primer tramo de recorrido, pero dado el esfuerzo realizado por el vehículo este presento fallas mecánicas y debió reducir su velocidad a la mitad con la que realizo el segundo tramo de recorrido y se mantuvo con esta velocidad constante hasta llegar a su destino. Se sabe que el vehículo inicio el viaje el día lunes a las 6:00 a.m. y llego a su destino el mismo día Lunes a las 3:00 p.m., durante el viaje no se realizó ninguna parada en ninguno de los trayectos y todos los tiempos y velocidades utilizadas se encuentran descritas en el problema, ¿cuál es la distancia que debió recorrer el vehículo para llegar de la ciudad de origen a la ciudad de destino?

a) 1040 kilómetros

b) 990 kilómetros

c) 900 kilómetros.

d) 1436 kilómetros.

Prueba técnica final:

1. Durante una reunión de los padres de familia de grado 11° se discute sobre la cantidad de alumnos que podría perder el año, para ello se observan estadísticas de los años anteriores que se presentan en la siguiente tabla:

Año	Porcentaje de alumnos de 11 que perdieron el año
2017	10%
2016	15%
2015	8%

Tabla 5: Pregunta 1 prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico

Un padre de familia afirma que el 100% de los estudiantes no llegara a graduarse este año, dado que en el 2017 1 de cada 10 estudiantes perdió el año y se sabe que se agruparon los estudiantes en 6 grupos iguales, pero su argumento es refutado por una madre de familia quien indica que la reducción del 5% de la mortalidad académica entre 2016 y 2017 es una constante que se presenta en el comportamiento de los alumnos que pierden el año, por lo que para el presente año hay pocas probabilidades de que pierda el año algún alumno.

Por su parte la rectora indica que para el primer periodo un 20% de los alumnos de grado 11° van perdiendo el año, si se sabe que la cantidad de alumnos de grado 11° de 2018 es igual que la cantidad de alumnos de grados 11° de 2017, que número de alumnos conforman el grado 11° de 2018 y cuantos van perdiendo el año?

- a) Son 55 y van perdiendo 15
- b) Son 60 y van perdiendo 6
- c) Son 50 y van perdiendo 10
- d) **Son 60 y van perdiendo 12**

2. En su lista de útiles escolares usted se encuentra con diversos útiles que debe comprar tales como cuadernos, lapiceros, libros, etc. Su presupuesto es limitado por lo que solo podrá comprar una parte de la lista, cada uno de los útiles de la lista posee un grado de prioridad asignado por el colegio, así como un precio estándar en el mercado de la siguiente manera:

Prioridad	Precio por Unidad	Útil
1	2000	Cuaderno
3	35000	Libro
1	1000	Lapicero
2	60000	Bolso
4	35000	Calculadora
3	2000	Regla
1	800	Lápiz
1	1500	Temperas

Tabla 6: Pregunta 2 prioridad, precio y utilidad; prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico

Nota: 1 es la máxima prioridad que puede tener cualquier útil.

El colegio le indica que los útiles que usted debe llevar según su grado son los siguientes:

Útil	Cantidad
Cuadernos	12
Lapiceros	4
Lápices	4
Bolso	1
Reglas	3

Tabla 7: Pregunta 2 útiles escolares prueba técnica final. Automatizar soluciones mediante pensamiento algorítmico

Si usted dispone de un presupuesto de 75.000 y teniendo en cuenta las prioridades dadas por el colegio cuales y cuantos útiles de cada tipo usted deberá comprar de forma que cumpla con su presupuesto y respete las prioridades establecidas.

- a) 12 cuadernos, 4 lapiceros, 4 lápices, 1 bolso, 3 reglas
- b) 1 bolso, 4 lapiceros, 4 lápices, 5 cuadernos
- c) 5 temperas, 12 cuadernos, 3 reglas, 4 lapiceros, 4 lápices
- d) 12 cuadernos, 4 lapiceros, 4 lápices, 3 reglas

- **Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos:** para evaluar esta habilidad se diseñaron preguntas que iban encaminadas hacia la descomposición de problemas.
- **Representar datos mediante abstracciones, como modelos y simulaciones:** También fueron usadas las preguntas de descomposición de problemas para evaluar este tipo de habilidad.

Prueba técnica inicial:

1. Como dividiría la siguiente situación, de forma que los grupos presentados ofrezcan la solución más óptima al problema.

Su colegio planea crear una panadería como proyecto productivo para que los estudiantes dediquen algo de tiempo libre a ella y puedan obtener alguna ganancia por el trabajo en dicha panadería. A los estudiantes de grado 11° se les asigna la labor de definir el proceso de producción de la panadería del colegio, se busca que el proceso sea lo más eficiente posible en términos de presupuesto y tiempo.

Un grupo de estudiantes se reúne a debatir acerca de cuál debe ser el proceso productivo de la panadería e identifican las siguientes fases clave:

1. Cotización de máquinas e insumos.
2. Estudio de mercado.
3. Compra de materiales para elaborar el pan (maquinaria e insumos).
4. Publicitar la inauguración.
5. Contratación de personal
6. Permisos de funcionamiento.
7. Inauguración de la panadería.
8. Hacer el pan.

Seleccione el orden en el cual se deben llevar a cabo los pasos anteriores con el fin de obtener el mejor resultado en términos de tiempo y presupuesto.

- a) 2,1,3,5,4,8,7,6
- b) 1,4,6,2,3,8,6,7
- c) 2,1,6,3,5,4,8,7
- d) 1,3,5,7,2,4,6,8

2. Usted ha ido a explorar a una montaña y se encuentra completamente solo, de repente comienza a llover y usted se refugia en una cueva que se hallaba en la montaña y rápidamente cae la noche, dadas las condiciones climáticas se oscurece muy rápido y se escucha el ruido de un animal grande ingresando a la caverna, usted lo ilumina rápidamente con su linterna y observa que se trata de un gran oso, rápidamente apaga su linterna y comienza a recordar las experiencias de supervivencia que ha observado en diferentes canales de televisión, recuerda que:

- los osos son animales con pésima visión nocturna y que incluso prefieren resguardarse antes que ir tras de una presa en la oscuridad.
- La forma más fácil de matar a un oso es disparar a uno de sus ojos y dada la velocidad del oso solo tendría una oportunidad de disparar.
- Los osos solo atacan cuando se sienten agredidos, por lo que dejarlos tranquilos es una buena idea.
- Dormir junto a un oso puede causar la muerte de una persona debido a que este podría aplastarlo en cualquier momento de la noche.

Usted revisa su bolso y observa que lleva un arma de fuego, suficiente comida para una semana y su teléfono el cual se encuentra sin señal, pero a los pocos metros sabe que encontrara un punto con cobertura. Cuál de las siguientes opciones escogería usted para salvar su vida.

- a) Sacar el arma de fuego y disparar al oso.
- b) Acomodarse en un rincón de la cueva durante la noche.
- c) escapar con sigilo de la cueva.
- d) Darle toda la comida al oso.

Prueba técnica final:

1. según la definición un número narcisista es un número de n dígitos, que coincide con la suma de las potencias n -ésimas de sus dígitos.

Por ejemplo:

153 es narcisista ya que: $1^3+5^3+3^3=153$.

1634 es narcisista ya que: $1^4+6^4+3^4+4^4=1634$.

Los primeros números narcisistas son: 1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407, 1634, 8208, 9474, 54748. etc.

Usted debe determinar si un número cualquiera X es o no narcisista, cuales son los pasos a seguir para determinar si dicho número es o no narcisista:

- a) Sumar cada dígito del número y posteriormente elevarlo a la 3 o la 4 y comparar que sea igual al número original.
- b) Sumar todos los dígitos de un número y a este resultado elevarlo a una potencia que sea igual al total de dígitos de dicho número y comparar con el número original.
- c) Elevar cada dígito a una potencia igual al total de dígitos de dicho número y luego sumar el resultado de cada operación, posteriormente comparar con el original.
- d) Ninguna de las anteriores

2. Durante la última década en la ciudad de Pamplona se han registrado fuertes lluvias, los entes de control del municipio se encuentran alarmados dado que los valores registrados indican que año a año las lluvias van en aumento, y en el año anterior se presentaron emergencias por derrumbes e inundaciones a lo largo de todo el municipio.

A continuación, se relaciona en una tabla los valores obtenidos durante los últimos 10 años de lluvias en Pamplona

Año	Cantidad de lluvias en mm de precipitación	Año	Cantidad de lluvias en mm de precipitación
2008	20	2013	40
2009	22	2014	45
2010	25	2015	50
2011	33	2016	58
2012	38	2017	67

Tabla 8: pregunta 2 prueba final. Formular problemas de manera que permitan usar computadoras y otras; Representar datos mediante abstracciones, como modelos y simulaciones

De acuerdo a la anterior información los entes municipales se proponen una solución radical para mitigar el impacto de las lluvias en el municipio, entre varias propuestas se escoge la de crear un domo desplegable que cubra el municipio de esta manera cuando la lluvia sea demasiada o muy fuerte con solo cerrar el domo se evitara el exceso de agua que provenga de la lluvia.

Se abre una convocatoria con el fin de recibir propuestas para el desarrollo de dicho proyecto, ¿en qué fases se debe descomponer un proyecto de tanta magnitud?

- a) Comprar los materiales, contratar la mano de obra, crear el domo, hacer los estudios correspondientes.

- b) Realizar los estudios correspondientes, tramitar los permisos necesarios, realizar presupuestos, compra de maquinaria, materiales y contratación de mano de obra, construir el domo.
- c) Conseguir todo lo necesario para la construcción del domo, gestionar todos los permisos necesarios, construir el domo.
- d) Todas las anteriores.

- **Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos:** Se diseñaron preguntas que comparaban diferentes procesos entre sí, así como preguntas sobre un proceso descrito, en su mayoría se usaron diagramas de flujo para la creación de estas preguntas.

- **Prueba técnica inicial:**

1. Cuál de los siguientes procesos es el que se asemeja más al de la figura.

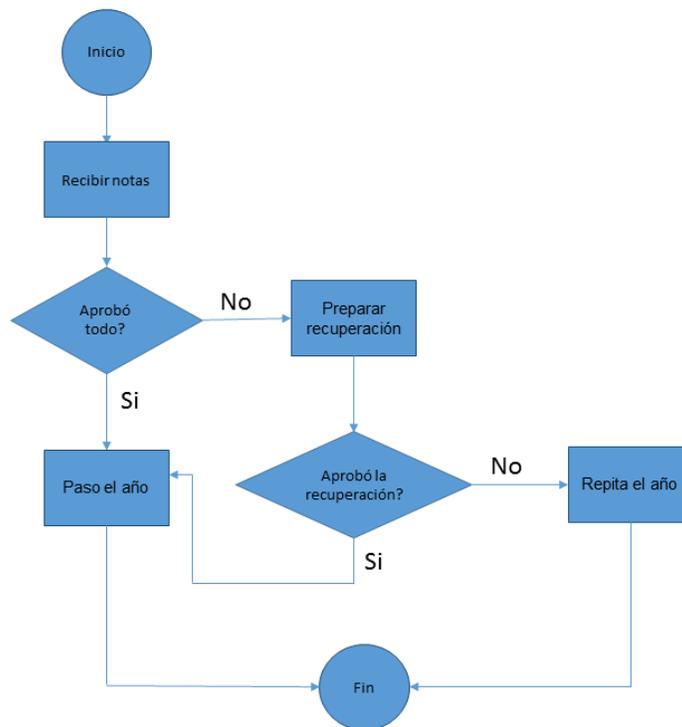


Ilustración 32: imagen de referencia pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

a)

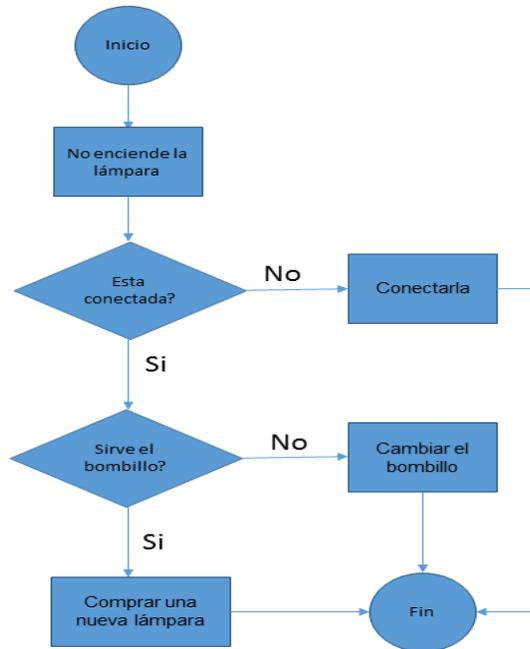


Ilustración 33: opción a pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

b)



Ilustración 34: opción b pregunta 1. prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

c)

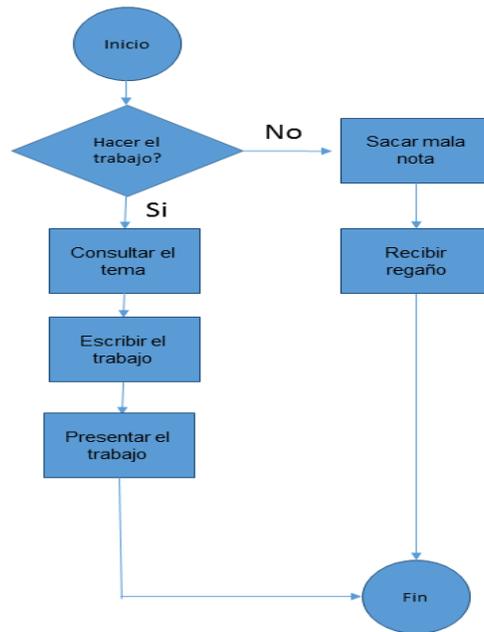


Ilustración 35: opción c pregunta 1, prueba técnica inicial. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

2. La gravedad es la fuerza que atrae los cuerpos entre sí, entre mayor sea la masa de un cuerpo, mayor será la fuerza de gravedad que este ejerce sobre los demás cuerpos, una persona que se encuentra en el planeta X es atraída hacia este con una fuerza 10 , pero dado que se trata de un astronauta esta persona debe dirigirse al planeta Y, antes de partir debe recibir un nuevo traje que resista 6 veces la gravedad del planeta X eso se hace con el fin de que los trajes ofrezcan un margen del 50% adicional de tolerancia a la gravedad de un planeta que esta por ser explorado. Según estos datos, ¿cuál es la gravedad que se espera encontrar en el planeta Y?
- a) 60
 - b) 120
 - c) 30
 - d) 15

- **Prueba técnica final:**

1. Observe el siguiente diagrama de flujo con cuidado y responda.

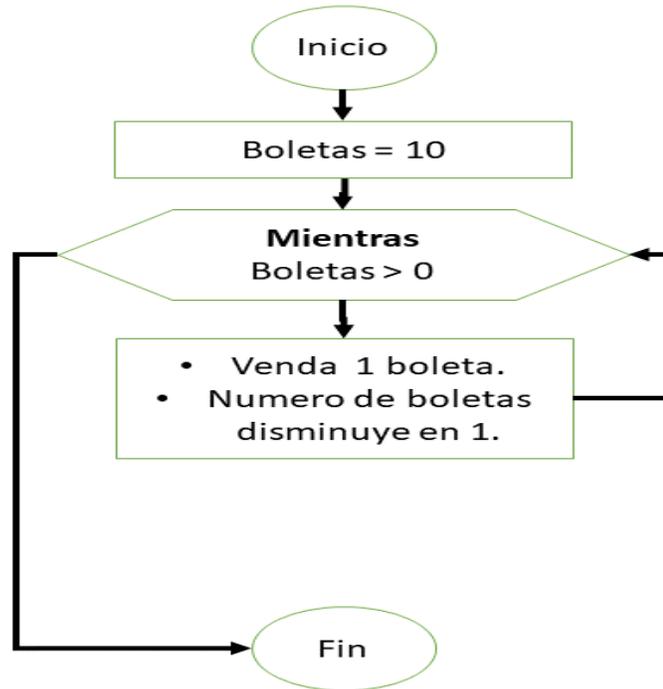


Ilustración 36: imagen de referencia pregunta 1, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

De acuerdo con la anterior instrucción, la instrucción



Es utilizada para:

- a) Indicar que se debe repetir la instrucción mientras
- b) Indicar el fin del diagrama
- c) Decir que no se deben vender más boletas
- d) Ninguna de las anteriores.

2. A continuación, se muestra una serie de instrucciones para ir de un punto A un punto B que se encuentra a 4 pasos de distancia, paso a paso se va indicando si la persona ya llegó o si por el contrario aún le queda camino por recorrer.

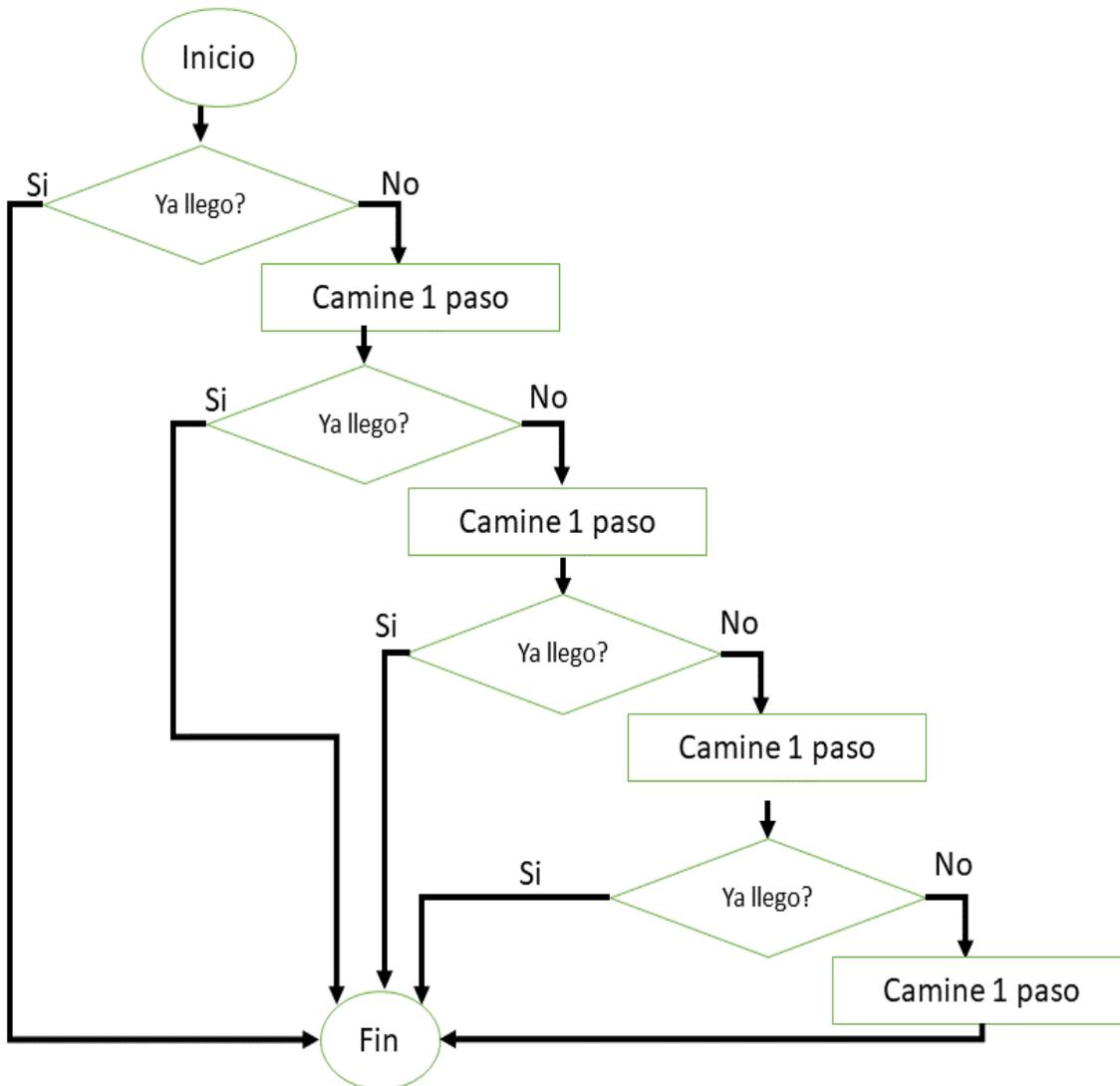


Ilustración 37: imagen de referencia pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

La forma correcta para que este código se pueda abreviar y funcione igualmente es:

a)

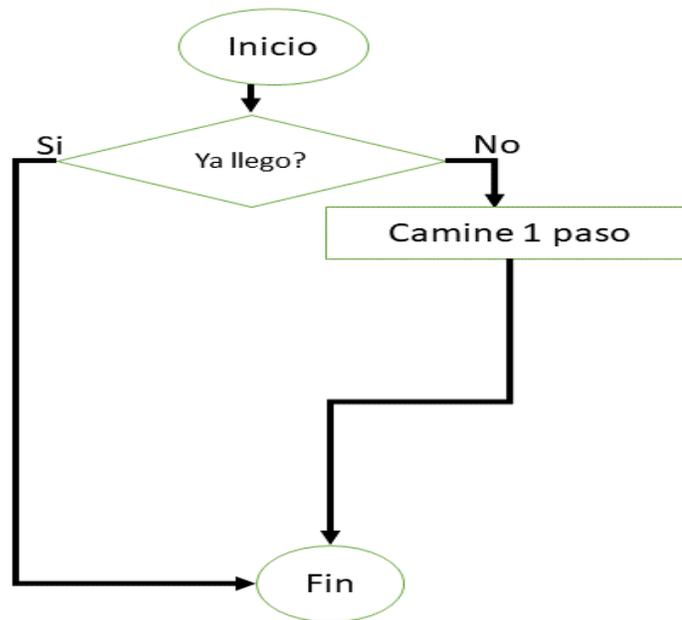


Ilustración 38: Opción a, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

b) esta

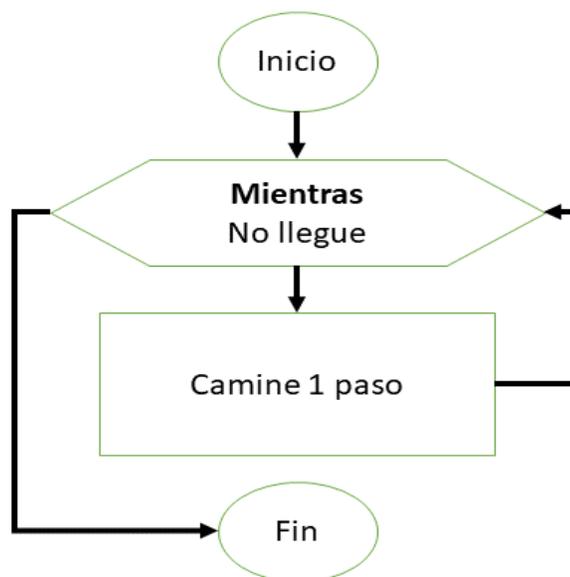


Ilustración 39: Opción b, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

c)

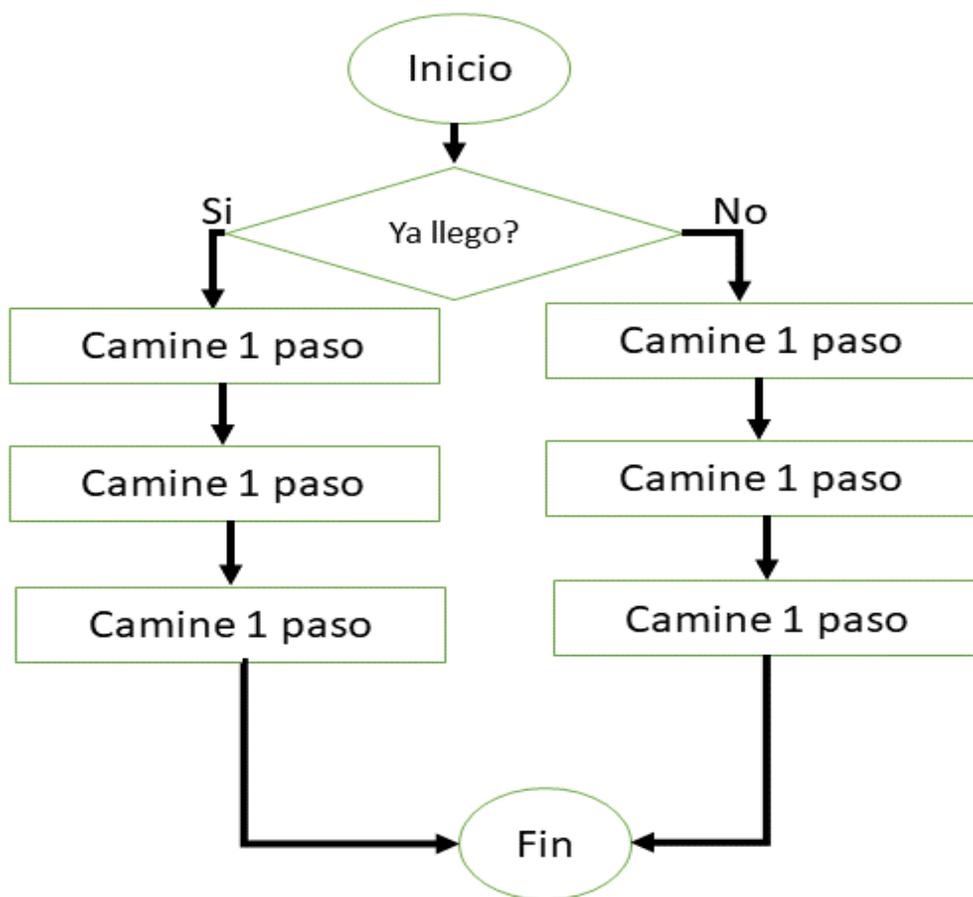


Ilustración 40: Opción c, pregunta 2, prueba técnica final. Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos

Fuente propia

5.5. Aplicación de la prueba técnica inicial

Para la aplicación de la prueba técnica inicial se utilizó la herramienta de formularios de Google, para facilitar la recolección y análisis de datos.

La prueba se aplicó en 2 sesiones, una para el grupo experimental y otra para el grupo de control.

La prueba del grupo experimental se aplicó el día 13 de marzo de 2018, iniciando a las 2:30 p.m., el último estudiante termino de presentar la prueba a las 3:35 p.m. del mismo día.

La prueba se aplicó al grupo de control el día 15 de marzo de 2018, comenzando a las 8:30 a.m., finalizo a las 9:45 a.m. cuando el ultimo estudiante que presento la prueba indico que ya había terminado esta.

La dirección donde se encuentra la prueba técnica inicial es la siguiente:

<https://goo.gl/forms/lCarSPupEIYQ1btL2>

5.6. Tabulación de los resultados de la prueba técnica inicial

Para el análisis de los resultados de la prueba técnica inicial se agruparon las preguntas por habilidades, se presentan dos graficas cada una correspondiente a los resultados obtenidos por cada uno de los grupos (grupo de control y grupo experimental).

En la prueba se han agrupado las habilidades de: formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos y representar datos mediante abstracciones, como modelos y simulaciones.

Con el objetivo de facilitar la visualización e interpretación del grafico obtenido, se han utilizado las siguientes convenciones:

- **H1:** Habilidad de Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.
- **H2:** Organizar datos de manera lógica y analizarlos.
- **H3:** Automatizar soluciones mediante pensamiento algorítmico.
- **H4:** Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos.
- **H5:** Representar datos mediante abstracciones, como modelos y simulaciones.
- **H6:** Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos.



Ilustración 41: Resultados prueba inicial del grupo experimental, por habilidades, en porcentajes

Fuente: imagen propia

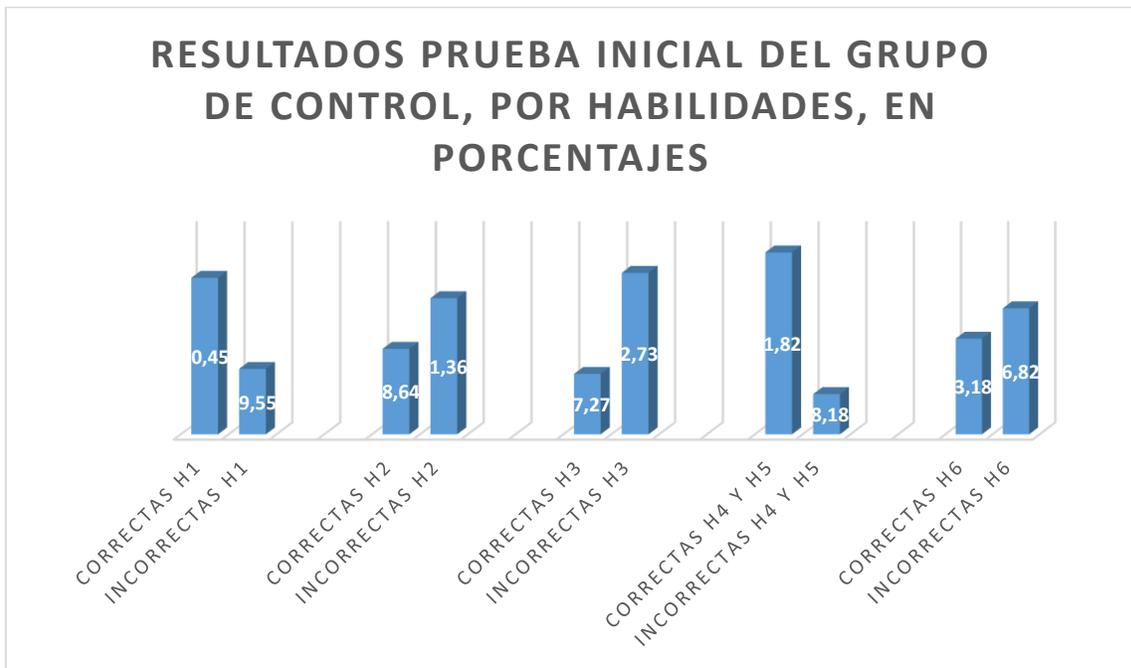


Ilustración 42: Resultados prueba inicial del grupo de control, por habilidades, en porcentajes

Fuente: imagen propia

Según los resultados obtenidos con la prueba técnica inicial, se presenta similitud en el rendimiento de los dos grupos, también fue posible observar en cuales habilidades se presentan bajos resultados y en cuales altos resultados, de la siguiente manera:

Habilidades con resultados positivos

Tanto la habilidad de formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos, como la habilidad de representar datos mediante abstracciones, como modelos y simulaciones, mostraron un alto rendimiento por parte de los dos grupos, cabe resaltar que estas dos habilidades son las únicas que se observan con resultados altamente positivos.

Habilidades con resultados promedio

La habilidad de identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva, obtuvo resultados promedio, dado que ambos grupos presentaron resultados por encima del 50% en cada una de las preguntas, pero no alcanzaron un porcentaje significativamente altos en ninguna de las preguntas que evaluaron esta habilidad.

Habilidades con resultados negativos

Se encontró debilidad en tres de las habilidades evaluadas, estas habilidades fueron:

- Organizar datos de manera lógica y analizarlos
- Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos
- Automatizar soluciones mediante pensamiento algorítmico

Los resultados obtenidos en la evaluación de estas tres habilidades son bajos, un patrón que se presentó en los grupos.

Al finalizar el análisis de la prueba técnica inicial se determina que los resultados obtenidos por los 2 grupos a lo largo de toda la prueba son muy similares, y las debilidades y fortalezas observadas se presentan en las mismas habilidades.

5.7. Asistencia al curso como observador y registro del comportamiento de los estudiantes

Durante el desarrollo del curso se tuvo apoyo de estudiantes del programa de ingeniería de sistemas quienes realizaban su trabajo social, estos fueron los encargados de orientar el desarrollo de las guías de trabajo y el manejo de la herramienta Scratch por parte del grupo de estudiantes.

Durante cada sesión se realizaron anotaciones, así como se llevó un control de las preguntas realizadas durante la sesión, muestras de entusiasmo en el trabajo, muestras de interés, entre otros indicadores.

Sesión N°1

Durante esta sesión se trabajó el manejo del programa Scratch con la ayuda de una guía llamada “Entendiendo el Escenario” ver anexo 1.

Observaciones del investigador:

En esta sesión inicialmente los estudiantes se mostraron atentos a las indicaciones iniciales, comienzan el desarrollo de la guía de manera ordenada y en silencio no presentan dificultades para la identificación del programa.

Se requirió mayor explicación para comprender el entorno de Scratch por lo que se requirió la explicación inicial de la guía

Se requirió la explicación de cómo configurar el lenguaje de Scratch.

La docente mostro interés en la herramienta y comenzó a desarrollar la guía al par de los estudiantes.

Los alumnos muestran interés por navegar en la aplicación de forma independiente, algunos no realizan las actividades propuestas por estar en otras actividades, pero todo el tiempo dentro de la aplicación.

Mediciones de la sesión:

- Preguntas realizadas durante la sesión:30 veces
- Muestras de entusiasmo en el trabajo:14 veces
- Muestras de interés:8 veces
- Muestras de apatía hacia el desarrollo de la sesión:6 veces
- Renuencia a seguir indicaciones del tutor:12 veces
- Interés en profundización en la temática:11 veces

Sesión N°2

En esta sesión se manejan las definiciones de variable y de constante, se utilizaron bloques de la herramienta que presentan un mayor grado de complejidad, debido a que los estudiantes ya tenían cierta familiaridad con esta proveniente de la sesión anterior. Ver anexo 2

Observaciones del investigador:

Al iniciar la sesión se presentó la dificultad del cambio de salón debido a adecuaciones en la parte eléctrica del salón en el que se trabaja regularmente. Se perdieron 45 minutos de la sesión en la adecuación del salón.

Los estudiantes dieron muestras de alegría mientras observaban que se iba a realizar el trabajo de la guía 2, lo que da para pensar que estuvieron esperando la segunda sesión.

En la primera actividad no se les permitió el uso del computador, se mostraron atentos a las indicaciones y definiciones dadas, se muestran más atentos que en la primera sesión.

Se muestran analíticos y tratan de realizar las actividades de la mejor manera posible.

Durante el trabajo los estudiantes comienzan con interés la actividad 2 hay algunas dudas, pero luego demuestran que realizan las actividades rápidamente antes del tiempo estipulado.

Demuestran dificultades en la orientación es decir los conceptos de derecha, izquierda y giros a la derecha y a la izquierda.

Hay dificultad para observar los enunciados debido a que el salón en el que se desarrolla el trabajo de hoy no es acorde, (Es un salón utilizado para el desarrollo de prácticas de mantenimiento de computadores).

Se mostró gran entusiasmo al iniciar el trabajo con Scratch por parte de los alumnos.

Se requirió guía especial para un alumno que no asistió a la 1 sesión por lo tanto este mostro dificultades para el manejo del entorno de la herramienta Scratch.

Para facilitar la revisión del trabajo fue necesario que los estudiantes realizaran los retos en Scratch sin pasarlos a las hojas.

Se tienen conceptos claros acerca de las diferencias entre variables y constantes.

Mediciones de la sesión:

- Preguntas realizadas durante la sesión: 22
- Muestras de entusiasmo en el trabajo: 14
- Muestras de interés: 10
- Muestras de apatía hacia el desarrollo de la sesión: 5
- Renuencia a seguir indicaciones del tutor: 8
- Interés en profundización en la temática: 15

Sesión N°3

Durante esta sesión se trabajó con el manejo de operadores lógicos y de operadores aritméticos, en esta sesión se presentó dificultad en comprender los conceptos de div y de mod. Ver anexo 3

Observaciones del investigador:

Al inicio de la sesión se muestra un grupo más tranquilo y con disposición para el trabajo, se observa que comienzan a comprender la metodología de trabajo que se desarrolla y aparentemente se hacen a una idea del nivel de seriedad y compromiso que se requiere para la correcta comprensión y desarrollo de las guías de trabajo de Scratch.

Muestran tener conocimiento sobre los operadores aritméticos a excepción del mod el cual con un sencillo ejemplo parecen que comprenden el concepto.

Se presenta gran habilidad en algunos alumnos para la resolución de problemas de este tipo, donde se debieron aplicar todas las operaciones se vio que algunos presentaron dificultades.

Se muestra algo de dificultad para asimilar el concepto de mod y sus diferencias con el div. Los alumnos que al inicio del curso se mostraban poco colaborativos se muestran más receptivos y colaboradores hacia el desarrollo de las diferentes actividades.

Luego del inicio del trabajo con operadores lógicos los estudiantes comenzaron a pensar sobre posibles casos en los cuales no se cumplan las condiciones.

El manejo de condicionales se dificulta debido a que no desean comprender las explicaciones, prefieren seguir con sus ideas sin aceptar las nuevas condiciones.

Mediciones de la sesión:

- Preguntas realizadas durante la sesión: 35
- Muestras de entusiasmo en el trabajo: 8
- Muestras de interés: 9
- Muestras de apatía hacia el desarrollo de la sesión: 5
- Renuencia a seguir indicaciones del tutor: 8
- Interés en profundización en la temática: 7

Sesión N°4

Previo al comienzo del desarrollo del trabajo en la guía 4 se realizó una prueba tipo Quiz a los alumnos por solicitud de la profesora encargada del grupo. En esta prueba se revisan los conocimientos teóricos adquiridos en el desarrollo de las anteriores sesiones. Ver anexo 4

Se desarrolló la temática de entrada de datos por teclado, así como se dio una primera aproximación al manejo de condicionales, se procuró integrar temáticas vistas de manera previa en esta guía como el manejo de constantes y de variables. Ver anexo 5

Observaciones del investigador:

Al inicio en la presentación del examen se mostró asimilación de los conceptos teóricos trabajados, algunos de los alumnos poseen dificultades en el manejo del div y del mod, se presentaron varios casos en donde se realizaron buenos diagramas de flujo que cumplían con todos los parámetros y condiciones estipuladas para la realización de esto

En el repaso se demostró que poseen los conceptos pero que se le presenta dificultad a la hora de aplicarlos a diferentes problemas de la realidad

Se muestra mayor fluidez en el manejo de la herramienta Scratch las preguntas dejan de orientarse sobre el manejo de la herramienta y comienza a orientarse sobre los ejercicios que se deben desarrollar, así como sobre la lógica que estos poseen

Mediciones de la sesión:

- Preguntas realizadas durante la sesión: 20
- Muestras de entusiasmo en el trabajo: 14
- Muestras de interés: 8
- Muestras de apatía hacia el desarrollo de la sesión: 4
- Renuencia a seguir indicaciones del tutor: 6
- Interés en profundización en la temática: 9

Sesión N°5

Para esta sesión se trabajó con el manejo de ciclos repetitivos, así como se profundizó en el manejo de la herramienta con la ayuda del cambio de escenarios y de disfraces que esta posee. Ver anexo 6

Observaciones del investigador:

En esta sesión se encontró dificultad para el trabajo debido a que solo se contó con 19 estudiantes esto dado a que los demás estudiantes del curso se encontraban en una actividad de la institución.

Comprenden que el uso de ciclos ayuda a reducir considerablemente la repetición de instrucción, se observa como utilizan los diferentes ciclos en cada una de las actividades propuestas en la guía, algunos notan rápidamente la diferencia entre los diferentes tipos de ciclos que ofrece Scratch lo que les permite escoger rápidamente el más adecuado según cada ejercicio a desarrollar lo amerite.

El manejo de la herramienta se nota cada vez más fluido, se nota asimilación en los conceptos y se hace claro que el trabajo realizado en la herramienta da sus frutos, se nota el interés y el entusiasmo de los estudiantes al ver las creaciones que son capaces de realizar.

Mediciones de la sesión:

- Preguntas realizadas durante la sesión: 12
- Muestras de entusiasmo en el trabajo: 19
- Muestras de interés: 11
- Muestras de apatía hacia el desarrollo de la sesión: 2
- Renuencia a seguir indicaciones del tutor: 2
- Interés en profundización en la temática: 3

Sesión N°6

En esta sesión se trabajó el uso de Scratch para la realización de animaciones, la idea era que los estudiantes aplicaran todo lo visto a lo largo del curso, y tuvieran una base para realizar el proyecto final que presentaron luego. Ver anexo 7

Observaciones del investigador:

Se retoman las sesiones luego de 15 días de ausencia debido a que a la profesora se le presentó una calamidad familiar y las clases debieron aplazarse, así mismo el proyecto se aplaza 8 días con el fin de poder evaluar completa la temática desarrollada en el curso mediante el proyecto.

Se muestra interés por la temática de las animaciones desarrolladas durante la sesión, los alumnos participan de manera activa y responden las preguntas realizadas por los tutores.

Algunos alumnos muestran que desarrollan las actividades más rápidamente que los otros, continúan con las actividades propuestas en la guía antes de que el tutor les indique, algunos combinan las actividades y las realizan a su propia manera.

El manejo de las coordenadas sigue dificultándose a los estudiantes, al proponerse cambios en el código se muestra dificultad.

Mediciones de la sesión:

- Preguntas realizadas durante la sesión: 25
- Muestras de entusiasmo en el trabajo: 18
- Muestras de interés: 15
- Muestras de apatía hacia el desarrollo de la sesión: 2
- Renuencia a seguir indicaciones del tutor: 2
- Interés en profundización en la temática: 17

Sesión N°7 (Proyecto final)

Como proyecto de final de curso los estudiantes debieron realizar en grupos de dos un proyecto de su propia inspiración que implementara todo lo trabajado en el curso, la temática era libre y los alumnos tenían la posibilidad de implementar el manejo de otros bloques diferentes a los trabajados en la clase.

A la hora de la revisión de los proyectos la sorpresa fue grata al encontrar proyectos con un grado de desarrollo alto, donde se demostró lo aprendido durante las sesiones, así como la creatividad e imaginación de cada uno de los grupos al presentar todas sus diferentes ideas.

Hubo algunos proyectos que demostraron poco tiempo e interés en su elaboración, pero mayoritariamente se mostraron proyectos con un buen nivel y donde los estudiantes demostraron manejo del mismo al responder las diferentes preguntas realizadas sobre la elaboración del proyecto y las dificultades encontradas.

A continuación, se muestra en una gráfica como fue la evolución a lo largo de las seis sesiones donde se desarrolló el material propuesto, de los ítems mencionados anteriormente.

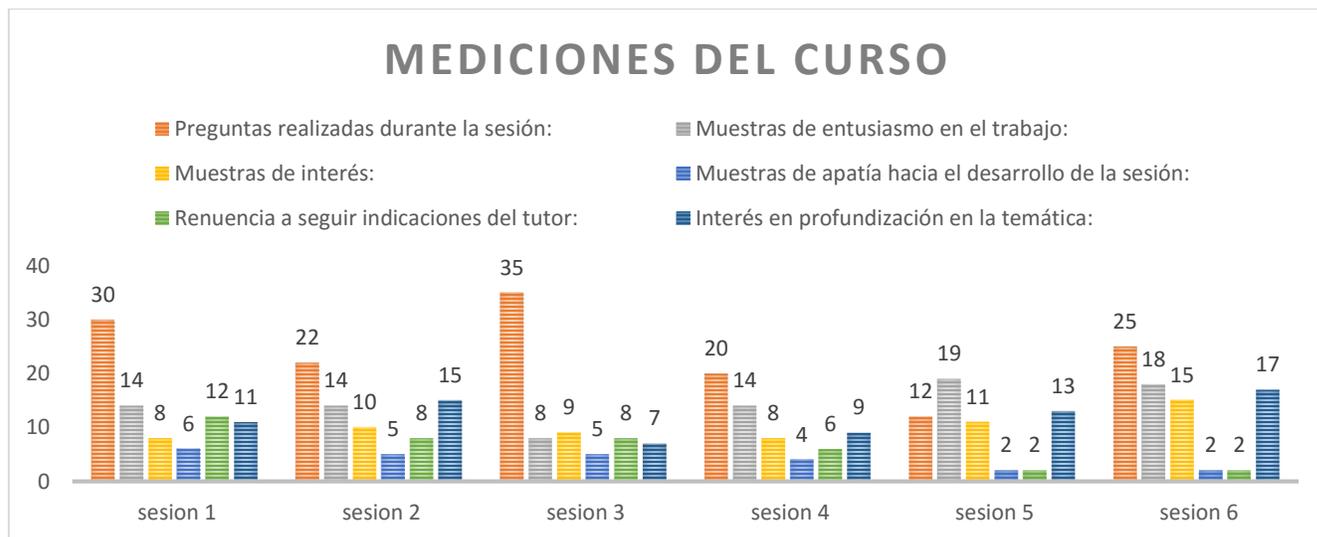


Ilustración 43: mediciones del curso

Fuente: imagen propia

5.8. Tabulación de resultados de la prueba técnica final.

Para el análisis de los resultados de la prueba técnica final se agruparon las preguntas por habilidades, se presentan dos graficas cada una correspondiente a los resultados obtenidos por cada uno de los grupos (grupo de control y grupo experimental).

En la prueba se han agrupado las habilidades de: formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos y representar datos mediante abstracciones, como modelos y simulaciones.

Con el objetivo de facilitar la visualización e interpretación del grafico obtenido, se han utilizado las siguientes convenciones:

- **H1:** Habilidad de Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.
- **H2:** Organizar datos de manera lógica y analizarlos.
- **H3:** Automatizar soluciones mediante pensamiento algorítmico.
- **H4:** Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos.
- **H5:** Representar datos mediante abstracciones, como modelos y simulaciones.
- **H6:** Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos.

RESULTADOS PRUEBA FINAL DEL GRUPO EXPERIMENTAL, POR HABILIDADES, EN PORCENTAJES

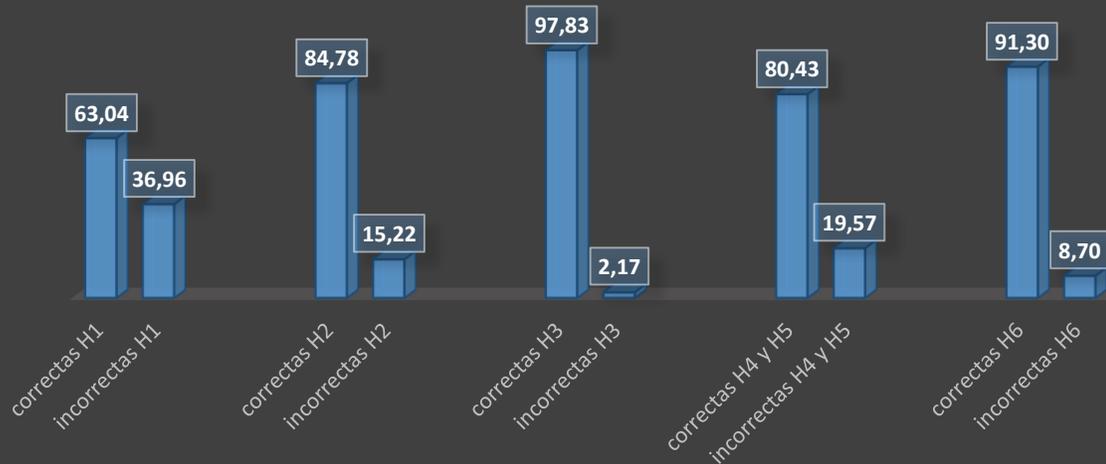


Ilustración 44: Resultados prueba final del grupo experimental, por habilidades, en porcentajes

Fuente: imagen propia

Resultados prueba final del grupo experimental, por habilidades, en porcentajes

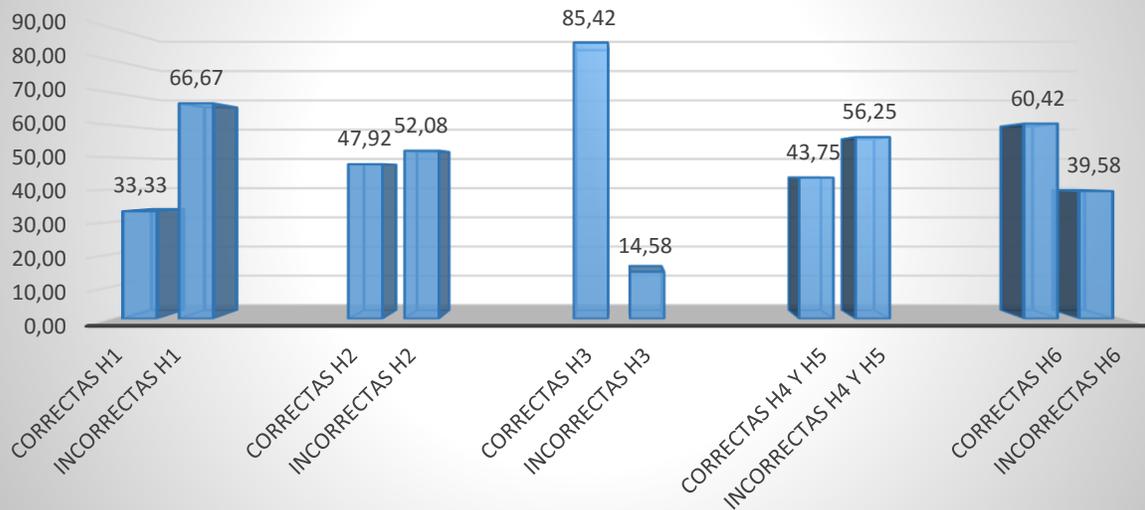


Ilustración 45: Resultados prueba final del grupo experimental, por habilidades, en porcentajes

Fuente: imagen propia

Se observa diferencia en los resultados de los dos grupos, el grupo experimental presenta resultados más positivos que el grupo de control en la habilidad **de identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.**

Los resultados del grupo experimental son bastante bajo en la habilidad de **Organizar datos de manera lógica y analizarlos**, nuevamente el grupo de control obtiene resultados positivos, que están bastante por encima de los del grupo de control.

Por su parte en la habilidad de **automatizar soluciones mediante pensamiento algorítmico** los resultados presentados por los dos grupos son buenos, se presentan mejores resultados en el grupo experimental.

En la habilidad de **generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos**, el desempeño del grupo de control fue bueno en promedio, por su parte el grupo experimental mostro grandes resultados en estas preguntas.

En las habilidades de:

- Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos.
- Representar datos mediante abstracciones, como modelos y simulaciones.

El desempeño del grupo de control fue bastante irregular, mientras que en el grupo experimental los resultados fueron positivos en términos generales, la segunda prueba se vio dominada por unos muy buenos resultados del grupo experimental mientras que el grupo de control presento resultados irregulares y en su mayoría negativos en la mayoría de las habilidades evaluadas.

5.9. Comparación de los resultados obtenidos en la prueba técnica inicial y la prueba técnica final

Los resultados obtenidos en las pruebas validan la teoría de desarrollo de las habilidades del pensamiento computacional con la ayuda de la programación.

en la primera prueba técnica (previa al curso), comportamiento de los dos grupos observados fue muy similar en su desempeño, las habilidades evaluadas presentaron un desempeño muy similar, lo que se explica dado que hasta el momento la formación de los dos cursos ha sido muy similar, ven las mismas materias con la misma intensidad horaria.

Para la segunda prueba técnica se presentó un incremento general en el desempeño de los resultados de la prueba final por parte del grupo experimental, el grupo de control presento un comportamiento similar en el desarrollo de las dos pruebas.

A continuación se describe la comparación de los resultados obtenidos en las dos pruebas técnicas, esta comparación se hará habilidad por habilidad evaluada con el fin de determinar en cuales el desarrollo fue mayor

- **Habilidad de Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva**

El grupo experimental se mantuvo en un desempeño promedio, mientras que el grupo de control presento un desempeño más bajo que el de la prueba técnica inicial, lo que supone que el grado de dificultad de la pregunta de la prueba técnica final afecto el desempeño del grupo de control pero no tuvo en mismo efecto en el grupo experimental quien mantuvo su desempeño a comparación de la prueba técnica inicial

- **Organizar datos de manera lógica y analizarlos**

Durante la prueba inicial los resultados en esta habilidad resultaron muy similares, en la prueba final el grupo de control bajo un poco su desempeño, mientras que el grupo experimental logro incrementar un poco su desempeño en esta habilidad

- **Automatizar soluciones mediante pensamiento algorítmico**

La primera prueba arrojó resultados muy bajos para esta habilidad, mientras que en la segunda prueba los resultados fueron por el contrario muy alto para ambos grupos, por lo que se determina que las pruebas sobre esta habilidad no son concluyentes.

- **Formular problemas de manera que permitan usar computadoras y otras herramientas para solucionarlos**

- **Representar datos mediante abstracciones, como modelos y simulaciones**

En estas habilidades el grupo de control arrojó resultados muchos menores en la prueba final que en la prueba inicial, mientras que el grupo experimental presentó un notable incremento en los resultados de estas habilidades.

Al parecer con el desarrollo del curso estas fueron las habilidades que se vieron más favorecidas al ser las que presentaron mayor incremento por parte del grupo experimental.

- **Generalizar y transferir el proceso de solución de problemas a una gran diversidad de estos**

Desde las pruebas iniciales se observó que esta habilidad se encontraba bien desarrollada, en la prueba final se observa como el grupo experimental obtiene resultados mejores que los del grupo de control.

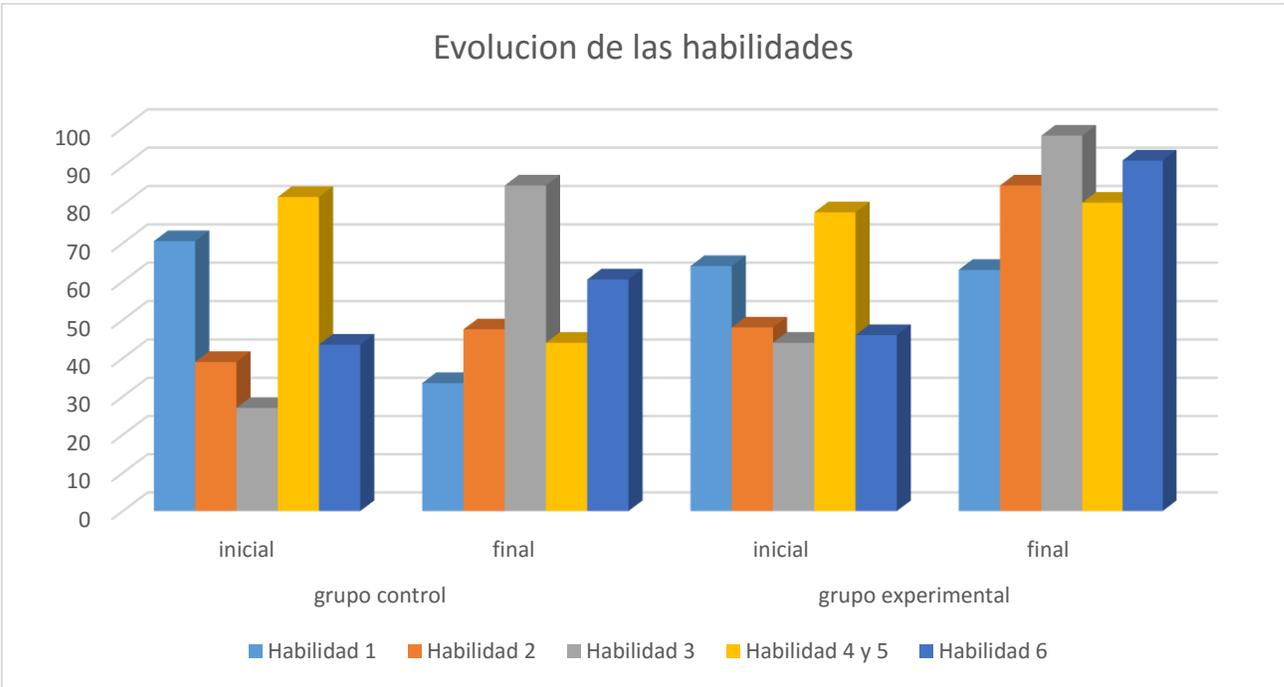


Ilustración 46: Evolución de las habilidades antes y después del curso

Fuente: Imagen propia

CAPITULO VI

CONCLUSIONES Y TRABAJOS FUTUROS

6.1. Conclusiones

- Se encontró que la institución educativa Bethlemitas Brighton, posee estudiantes que tiene grandes habilidades y características para el aprendizaje de la programación, también se resalta la colaboración y buena disposición de los docentes y administrativos de la institución hacia el desarrollo del proyecto, dado que colaboraron en todos los aspectos logísticos posibles según sus capacidades.
- Al inicio del curso, los estudiantes poseían cierto nivel de desempeño en cada una de las habilidades evaluadas, el cual era aceptable, este nivel lo habían alcanzado gracias al desarrollo de sus clases normales, se evidencio que algunos estudiantes poseen mayor desarrollo de sus habilidades que otros.
- Durante el desarrollo del curso fue grato observar el grado de interés y compromiso que asumieron los estudiantes, fue notorio que entre más tiempo usaban la herramienta Scratch mayor era su comprensión de la misma lo que les facilitaba la realización de algunos ejercicios.
- Con la evaluación del proyecto final fue claro el nivel que se logró alcanzar, así como el grado interés mostrado por la gran mayoría de participantes del curso, esto se vio reflejado en la entrega de muy buenos proyectos donde se ve el manejo y comprensión de las temáticas desarrolladas.
- Para la obtención de resultados mayores se requiere un mayor tiempo de trabajo con los estudiantes así como la adquisición de materiales didácticos que permitan llevar la computación a un nivel más allá de la programación.
- Fue claro en la comparación entre las dos pruebas técnicas y los dos grupos, que las habilidades que se buscaban desarrollar se vieron ampliamente favorecidas en el grupo experimental, lo que valida la hipótesis que para desarrollar el pensamiento computacional y sus habilidades la programación es una herramienta de gran ayuda.

6.2. Trabajos Futuros

Se recomienda que este proyecto se desarrolle más, implementando nuevas técnicas y herramientas para el desarrollo del mismo, los resultados aquí obtenidos pueden ser mejorados y reevaluados en futuros trabajos.

A futuro se pueden utilizar diferentes metodologías de aprendizaje como las E-Learning que permitan observar como en un ambiente virtual los estudiantes tienen la posibilidad de adquirir nuevos conocimientos y desarrollar las habilidades aquí descritas. A su vez sería pertinente comparar los resultados de un curso que se desarrolle bajo la metodología E-Learning con los resultados de este curso que se desarrolló bajo la metodología presencial con el apoyo de guías didácticas.

También se plantea la posibilidad de usar otro tipo de herramientas como Legos, u otros lenguajes de programación diferentes a Scratch con el fin de comparar los resultados de estos con los aquí descritos y encontrar cuál de estas herramientas es la que más ayuda en el desarrollo del pensamiento computacional.

REFERENCIAS

Bibliography

- Alberto Gómez, & Chacón Larraz. (2016). Instalación de Scratch Jr en un PC. Recuperado el 23 de mayo de 2018, a partir de <http://codigo21.educacion.navarra.es/2016/01/29/instalacion-de-scratch-jr-en-un-pc/>
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kamylyis, P., & Punie, Y. (2016). Developing Computational Thinking : Approaches and Orientations in K-12 Education. *Proceedings EdMedia 2016*, (June), 1–7. <https://doi.org/10.2791/792158>
- Capot, R. B., & Espinoza, R. M. (2015). Desarrollo del Pensamiento Computacional con Scratch, 616–620. Recuperado a partir de <http://www.tise.cl/volumen11/TISE2015/616-620.pdf>
- Carro, M., Dpto, R., & Villamil, L. (2009). Programa winlogo.
- Date, I. (2018). La habilidad para solucionar problemas y el aprendizaje de lenguajes computacionales en el ámbito de la educación básica : un estudio cualitativo con alumnos de sexto grado-Primera edición.
- Del, A., Aprendizaje, P. D. E., López, M. F., & Olmo, P. Del. (s/f). Resolución de problemas computacionales: análisis del proceso de aprendizaje.
- Department for Education. (2013). Physical education Programmes of Study: Key Stages 3 and 4. *Physical education programmes of study: key stages 3 and 4*, (September 2013), 3. <https://doi.org/FE-00190-2013>
- EDUTEKA. (2012). Eduteka - Programación en la Educación Escolar & Scratch & Recursos. Recuperado el 23 de mayo de 2018, a partir de <http://eduteka.icesi.edu.co/modulos/9/285/2088/1>
- Gonz, M. (2013). Pensamiento computacional, 2–6. Recuperado a partir de <http://www.pensamientocomputacional.org/index.php/home/menu-definicion>
- Huellas, F. (2010). Nuestra Historia. *Asorut*, 4. Recuperado a partir de <http://bethlemitasbrightonpamplona.edu.co/nuestra-historia/>
- IPE Buenos Aires, & Ministerio de educación de la Nación - Argentina. (2000). Módulo 7 - Resolución de problemas. En *Competencias para la profesionalización de la gestión educativa* (p. 37).
- Ingeniería de Sistemas - Unipamplona - Pensamiento Computacional. (s/f). Recuperado el

28 de mayo de 2018, a partir de

http://www.unipamplona.edu.co/unipamplona/portaIG/home_77/recursos/01general/06022018/02_pensamiento.jsp

Juan Antonio Pascual. (2015). Scratch, programación sencilla y gratis para niños y mayores | Tecnología - ComputerHoy.com. Recuperado el 23 de mayo de 2018, a partir de <https://computerhoy.com/noticias/software/scratch-programacion-sencilla-gratis-ninos-mayores-37925>

León Vásquez, A. (s/f). Desarrollo del pensamiento computacional por medio de la creación de objetos virtuales de aprendizaje con Scratch por los estudiantes de media técnica para la educación básica primaria.

Oficina de Planeación - Universidad de Pamplona - Sistemas de Información. (s/f).

Recuperado el 28 de mayo de 2018, a partir de

http://www.unipamplona.edu.co/unipamplona/portaIG/home_71/recursos/01_general/11042013/sistemas_informacion.jsp

Pérez, M. (2017). EL PENSAMIENTO COMPUTACIONAL PARA POTENCIAR EL DESARROLLO DE HABILIDADES RELACIONADAS CON LA RESOLUCIÓN CREATIVA DE PROBLEMAS. *3C TIC*, 6(Edición 20), 38–63.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... Kafai, Y. (2009). Scratch: Programming for All. *Communications of the ACM*, 52, 60–67. <https://doi.org/10.1145/1592761.1592779>

Rincón Rueda, A. I., & Ávila Díaz, W. D. (2016). Una aproximación desde la lógica de la educación al pensamiento computacional. *Sophía*, 2(21), 161. <https://doi.org/10.17163/soph.n21.2016.07>

Sánchez Ruiz, L. A. (2016). COMPRENDIENDO EL PENSAMIENTO COMPUTACIONAL: EXPERIENCIAS DE PROGRAMACIÓN A TRAVÉS DE SCRATCH EN COLEGIOS PÚBLICOS DE BOGOTÁ.

Urbina Ramírez, S. (1999). Informática y Teorías del Aprendizaje. *Pixel-Bit: revista de medios y educación*, 12, 87–100.

Valverde-Berrocoso, J., Fernández-Sánchez, M. R., & Garrido-Arroyo, M. C. (2015). El pensamiento computacional y las nuevas ecologías del aprendizaje. *Revista de Educación a Distancia (RED)*, (46). <https://doi.org/10.6018/red/46/3>

Wing, J. M. (2006). Wing06-ct, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>

Yoana, L., & Gómez, G. (2014). COMPETENCIAS MÍNIMAS EN PENSAMIENTO COMPUTACIONAL QUE DEBE TENER UN ESTUDIANTE ASPIRANTE A LA

MEDIA TÉCNICA PARA MEJORAR SU DESEMPEÑO EN LA MEDIA TÉCNICA
DE LAS INSTITUCIONES EDUCATIVAS DE LA ALIANZA FUTURO DIGITAL
MEDELLÍN.

Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital.
Revista de Educación a Distancia (RED), (46). <https://doi.org/10.6018/red/46/4>

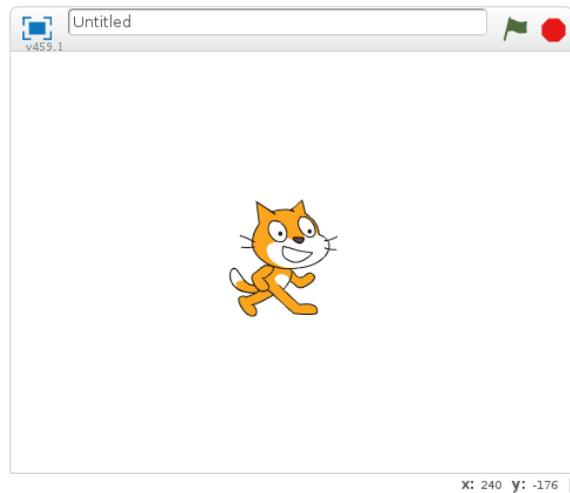
ANEXOS

Anexo 1

Guía 01

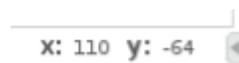
Entendiendo el Escenario

Las aplicaciones que se desarrollan en Scratch tienen en común el escenario, en el cual se lleva a cabo la ejecución del programa



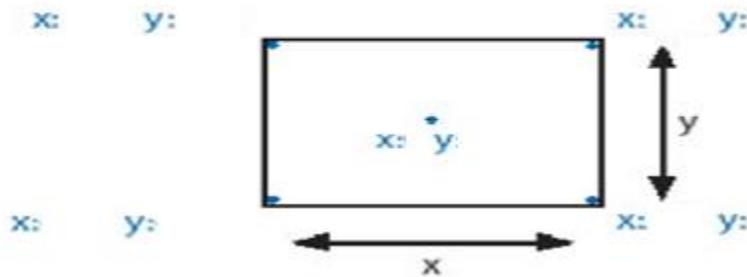
Referencia 1

Es importante entender la forma y las coordenadas de un escenario, mueva el mouse por la superficie del escenario y observe las coordenadas X Y



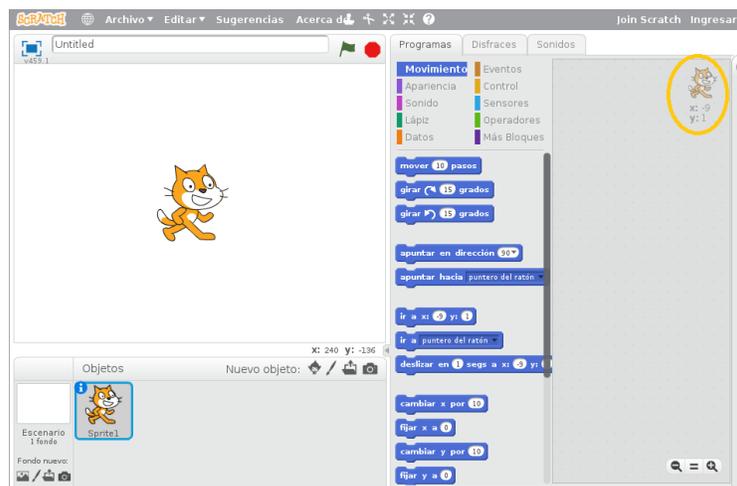
Actividad 1

Determine los valores X Y para cada esquina de escenario



Entendiendo al personaje

Por defecto todos los proyectos nuevos en Scratch tienen un único objeto, el objeto Gato. El Gato se encuentra ubicado en una posición X Y en el escenario se puede observar en la esquina superior derecha X Y



Mueva el Objeto Gato con un clic sostenido por el escenario y observe la posición X Y donde se encuentra el personaje.

Todo tiene un inicio

En la parte superior del escenario se observan dos botones



Inicia la ejecución de los programas en el proyecto.



Detiene la ejecución de los programas del proyecto.

Para poder usar estos botones es necesario utilizar al principio de la programación un bloque



de tal forma que cuando se pulse  se ejecuten las instrucciones que van a continuación del bloque.

Actividad 2

Mover el personaje Gato de la esquina superior Izquierda del escenario unos pasos, realizando un saludo Hola Bienvenidos al curso durante el movimiento.

¿Que pasos son necesarios para realizar la actividad 2?

- 1.
- 2.
- 3.
- 4.
- 5.
- 6.

Compare los pasos que definió con los siguientes:

- ✓ Primero sería crear un bloque al presionar bandera que indicaría el inicio de la ejecución del programa.
- ✓ Segundo Ubicar al personaje Gato en la esquina superior izquierda.
- ✓ Tercero sacar un mensaje que diga Hola y que dure N segundos
- ✓ Cuarto Mover el personaje una cantidad de pasos a la derecha.
- ✓ Quinto sacar un mensaje que diga Bienvenidos al Curso que dure N segundos.

Desarrollar el Código:

1. Ubique la pestaña Programas en la categoría Eventos y arrastre el bloque



al área de programación.

2. Ubique la pestaña Programas en la categoría Movimiento y arrastre el bloque



al área de programación encajándolo debajo del bloque que inserto en el paso 1. Asigne en X Y los valores que ubiquen al gato en la esquina superior izquierda.

3. Ubique la pestaña Programas en la categoría Apariencia y arrastre el bloque



al área de programación encajándolo debajo del bloque que inserto en el paso 2. Asigne Hola y el tiempo 0.8 segundos

4. Ubique la pestaña Programas en la categoría Movimiento y arrastre el bloque



al área de programación encajándolo debajo del bloque que inserto en el paso 3. Asigne la cantidad de pasos (Numero entero positivo o negativo) que debe dar el Gato.

5. Ubique la pestaña Programas en la categoría Apariencia y arrastre el bloque



al área de programación encajándolo debajo del bloque que inserto en el paso 4. Asigne Bienvenidos al Curso y el tiempo 1 segundo

Al final debe tener la siguiente estructura:



Ejecute el programa usando el control Bandera Verde

Referencia 2:

Un algoritmo es una serie de pasos ordenados, detallados y finitos que permiten realizar un determinada actividad.

Reto 1:

Termine el movimiento del Gato hasta que llegue al extremo del escenario, puede mostrar

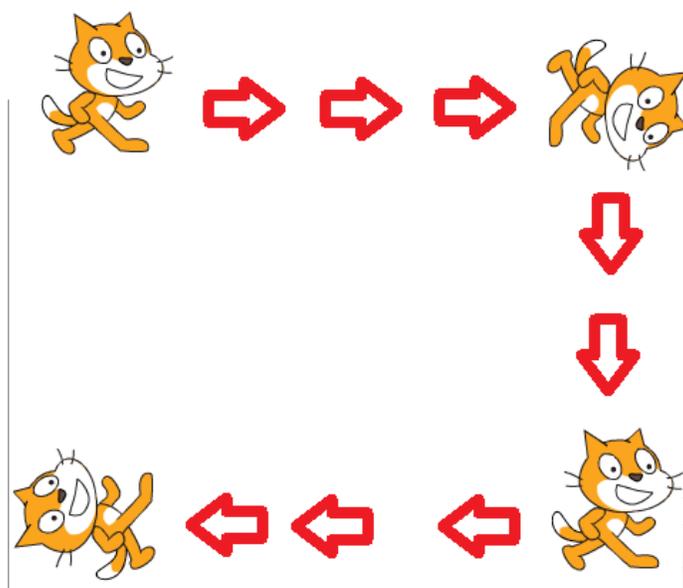


otros mensajes o usar el bloque necesarias.

para realizar las pausas

Reto 2:

Realice el siguiente movimiento del personaje gato por el escenario.

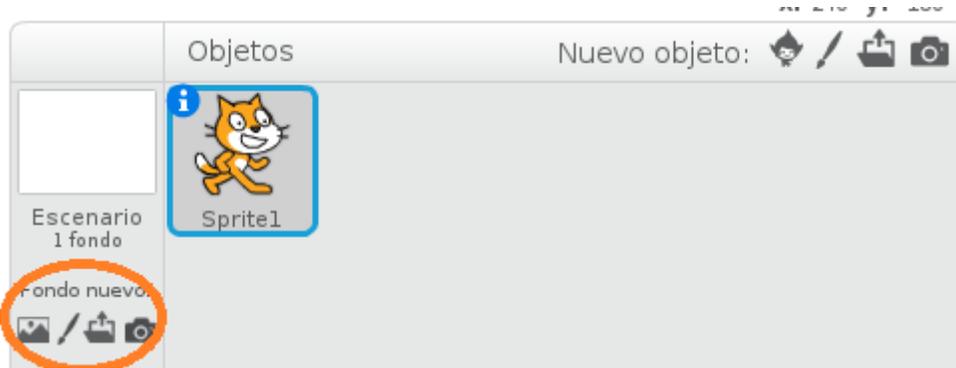


Puede ser necesario probar el funcionamiento de los siguientes bloques.



Actividad 3

Por defecto todo nuevo proyecto tiene fondo blanco en el escenario, para cambiarlo se usa cualquiera de las siguientes opciones:

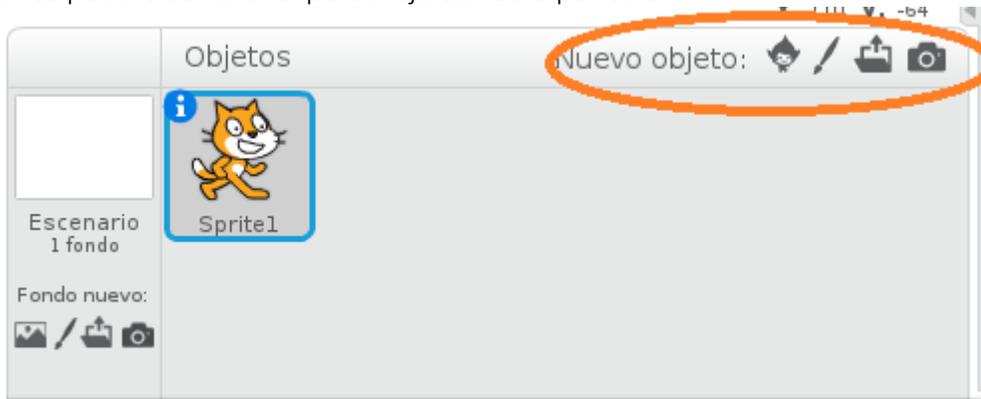


Para esta actividad vamos a cargar un fondo dado por su tutor con la imagen de un laberinto.

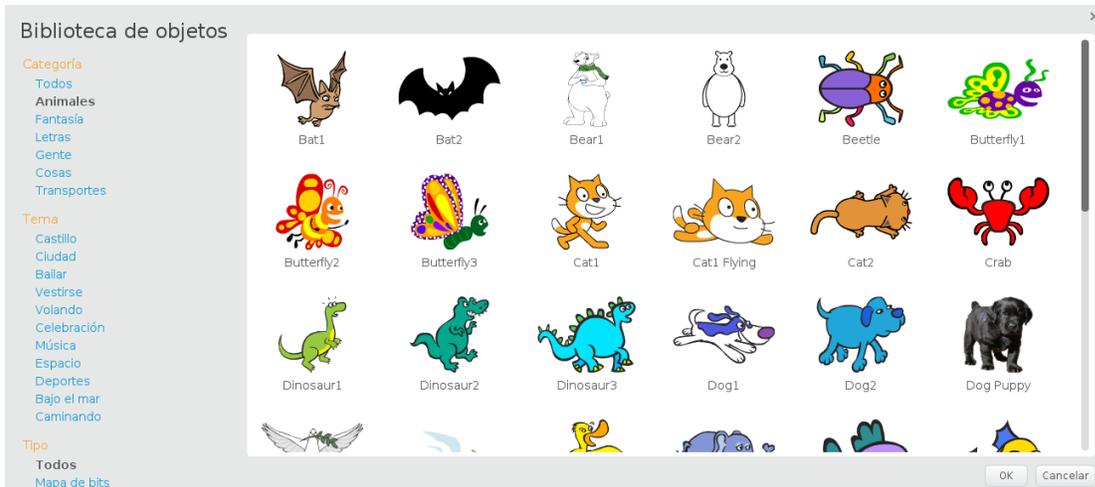
1. Clic en 
2. Seleccionar la imagen en la ruta donde se encuentre.
3. Puede eliminar el fondo Blanco haciendo clic en la X



También es posible cambiar el personaje del Gato por otro.



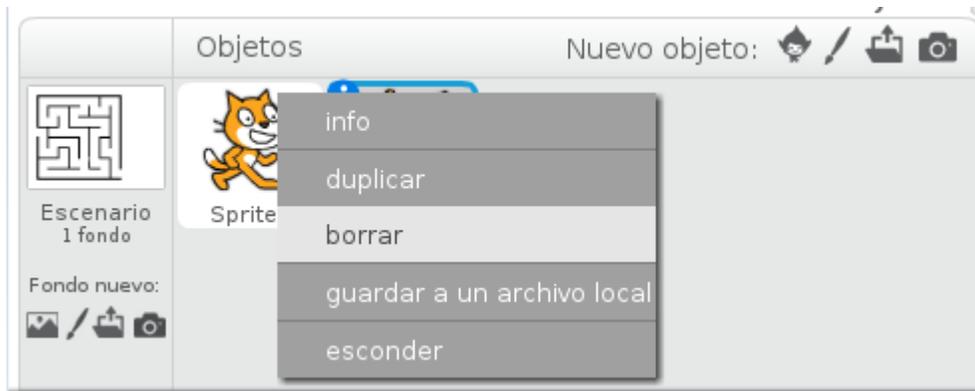
1. Clic en 
2. Ubique en la categoría Animales el Beetle



3. Reduzca el tamaño del personaje con la herramienta haciendo clic's consecutivos.

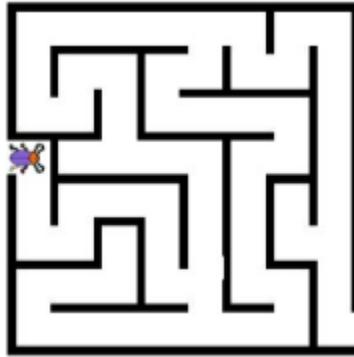


4. Elimine el Gato clic derecho sobre el gato y Borrar.



Reto Final

Realice el recorrido del escarabajo por el laberinto donde salga al otro extremo.



Compare con sus compañeros y determinen quien lo hizo más real y uso menos instrucciones.

Anexo 2

Guía 02

Recapitulando la guía anterior

A que hace referencia la siguiente definición:

“Conjunto definido, ordenado y finito de pasos que permite realizar una tarea”

Donde:

Ser definido: Sin ambigüedad, cada paso debe indicar la acción a realizar sin criterios de interpretación.

Ser finito: Un número específico y numerable de pasos, el cual deberá finalizar la realización de la tarea al completarlos.

Actividad 1

Con ayuda de una hoja de papel asigne la numeración correcta a los siguientes pasos para lograr el objetivo de “Lavarse las manos”.

- ___ Tomar el Jabón con la mano derecha
- ___ Girar la llave a la izquierda
- ___ Pararse enfrente del lavamanos
- ___ Mojarse las manos
- ___ Mover el jabón de mano a mano hasta conseguir abundante espuma.
- ___ Girar la llave a la derecha
- ___ Mojar el Jabón
- ___ Retirar la espuma de las manos frotándolas debajo del agua.
- ___ Secarse las manos usando una toalla.

Reflexione:

¿Son todos los pasos? ¿Hizo falta algún paso? ¿Cuál? ¿Sobra algún paso?

Actividad 2

Diseñe un algoritmo que permita a su tutor que está parado en **Posición** salir del salón de clase.

Puede Usar las Instrucciones

- ✓ Camine X Pasos
- ✓ Gire 45°

Referencia 1

Cuando se analiza un problema existirán datos de entrada que son los valores con los que se inicia el análisis, posteriormente al desarrollar la lógica de solución al problema se generan nuevos datos que deben ser controlados y almacenados para lograr así encontrar la solución. Estos datos se pueden clasificar en:

- ✓ **Variables:** Es una característica de una entidad que puede adquirir diferentes valores durante su periodo de vida.
- ✓ **Constante:** Es una característica de una entidad que no puede adquirir diferentes valores durante su periodo de vida.

De los siguientes ejemplos determine cuáles son variables y cuales son constantes.

N°	Nombre	Variable	Constante
1	La edad de una persona		
2	El valor de π		
3	La tasa cambio del dólar		
4	El clima de su ciudad		
5	La estatura de una persona		
6	Los padres biológicos de una persona		
7	Las unidades de Medida Cm Mtr Km etc		

De forma genérica en programación se denominan variables que son contenedores capaces de almacenar información.

Actividad 3

Convertir la temperatura 36.4° Centígrados a su equivalente en grados Fahrenheit

Es conveniente que cada estudiante se realice las siguientes preguntas:

- ¿Qué conocimientos tengo en el área o áreas del problema?
- ¿Son suficientes esos conocimientos?
- ¿Dónde puedo obtener el conocimiento que necesito para resolver el problema?
- ¿Mis compañeros de estudio me pueden ayudar a clarificar mis dudas?
- ¿Qué expertos en el tema puedo consultar?

Análisis del problema:

- ¿Qué solución se debe entregar?
- ¿Qué datos Constantes y variables conozco?
- ¿Que está prohibido hacer?
- ¿Que está permitido hacer?
- ¿Qué formulas debo emplear?
- ¿Qué pasos debo seguir para obtener la solución?

Análisis del Problema

¿Que solución se debe entregar?



Temperatura
en Fahrenheit

¿Qué datos Constantes y Variables conozco?



Temperatura
en Centígrados

Nada

¿Que está prohibido hacer?

Todo

¿Que está permitido hacer?

$$F = \left(\frac{9}{5} * C\right) + 32$$

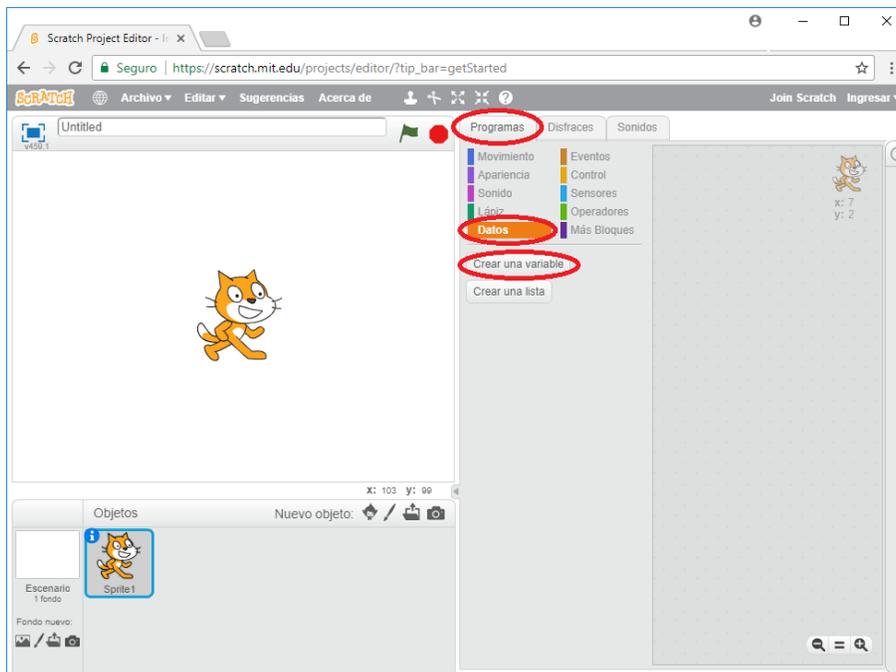
¿Qué formulas debo emplear?

1. Almacenar en una variable la temperatura en grados Centígrados “TemCen”
2. Aplicar la formula de conversión y almacenar el resultado en una Variable “TemFah”

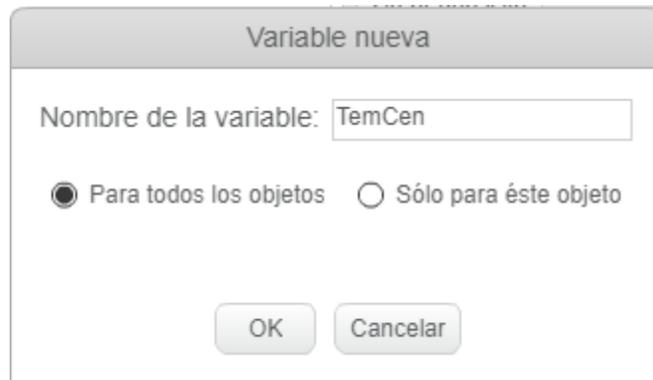
¿Qué pasos debo seguir para obtener la solución?

Traducir a Scratch

- 1) Inicie un proyecto en Scratch
- 2) Ubique la pestaña Programas en la categoría Datos y clic en crear una variable.



- 3) En Nombre de la variable ingrese “TemCen” y clic en Ok



Observe los diferentes bloques que se crearon para el control de la variable



- 4) Cree la variable “TemFah”
- 5) Una vez preparados los contenedores de los datos (variables) se puede dar inicio a la programación de los bloques.
- 6) Todo tiene un inicio... Cree el bloque “**al presionar**” para iniciar el programa.



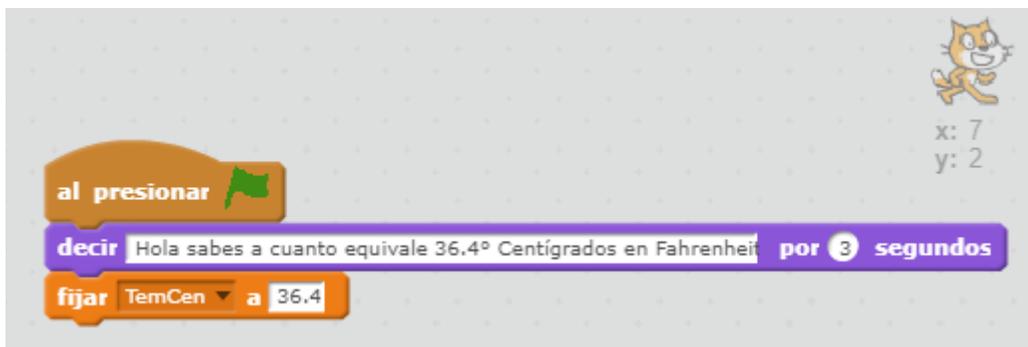
- 7) Seria interesante que el personaje Gato nos comentara que va a ocurrir agreguemos un mensaje donde nos diga “Hola sabes a cuanto equivale 36.4° Centígrados en Fahrenheit” y que dure 3 segundos el mensaje.



- 8) Ahora demos asignar en la variable “TemCen” el valor 36.4. Ubique la categoría Datos y utilice el bloque Fijar, seleccione la variable TemCen y asigne el valor.



Debe quedar así:



- 9) Debemos aplicar la formula en un conjunto de bloques y fijarla en la variable “TemFah”

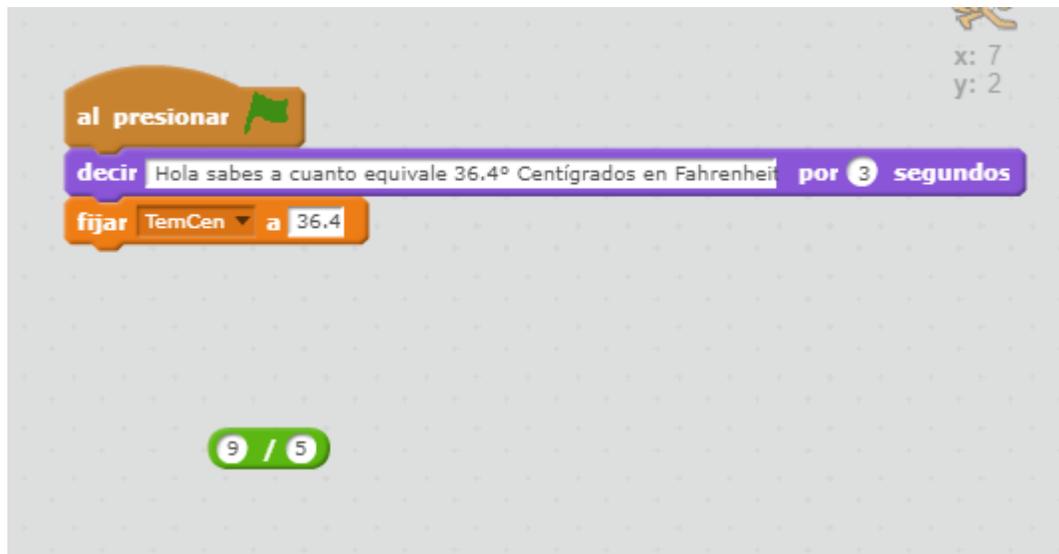
La formula se debe descomponer en partes y posteriormente unirla en una sola.

$$F = \left(\frac{9}{5} * C\right) + 32$$

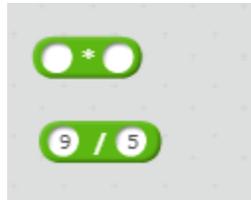
Ubique la categoría “Operadores” y realice los siguientes pasos.



1. Realizar 9/5



2. Multiplicar 9/5 por TemCen
a. Añadir un bloque de Multiplicación



b. Añadir la variable TemCen desde la categoría Datos



c. Arrastrar el bloque de división y el de la variable TemCen al bloque de Multiplicación.



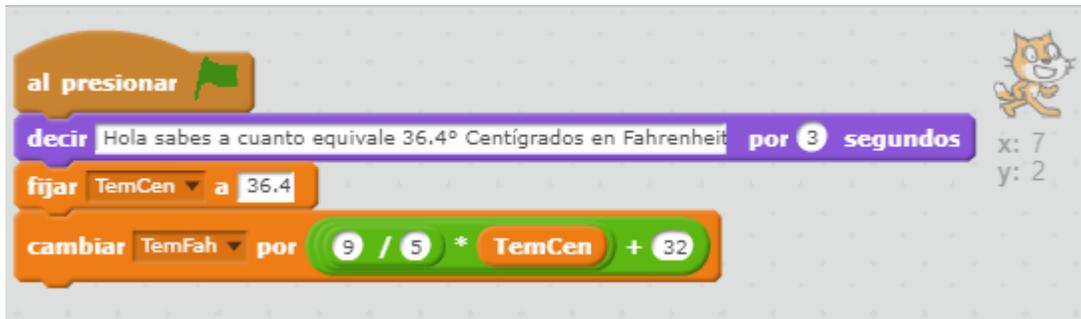
3. Agregar un bloque suma para finalizar la formula



4. Asigne el resultado de la formula a la variable TemFah con ayuda del bloque "Cambiar" que están en la categoría Datos.



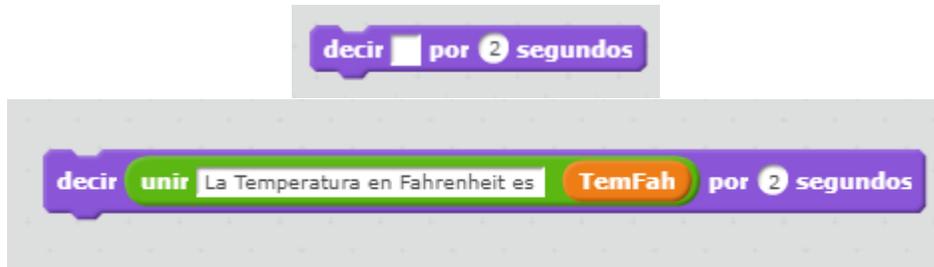
10) Acople el bloque al programa principal.



11) Genere un mensaje del Gato para imprimir la respuesta. Para esto debemos formar la respuesta “La Temperatura en Fahrenheit es” con el valor de la variable “TemFah”, utilizaremos de la categoría Operadores el bloque Unir



Ahora usamos un bloque decir



12) Se une este ultimo bloque con el programa principal.



Reto 1:

Cambio el problema ahora conoce la temperatura en grados Kelvin 45.3° y debe convertirla a grados Centígrados y grados Fahrenheit.

Reto 2:

Un viajero quiere convertir su dinero en Dolares y Euros de la siguiente forma 60% en dólares y 40% a Euros el cuenta con 2'500.000 pesos. Realice los análisis necesarios para lograr la solución.

Reto 3:

Invente su propio problema y compártalo con sus compañeros.

Anexo 3

Guía 03

Referencia 1

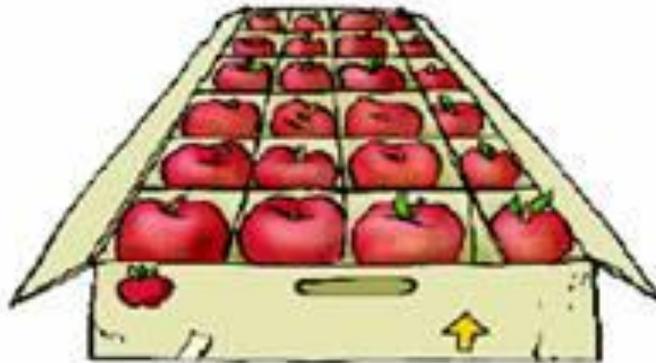
Cuando se resuelve un problema es necesario en algunas ocasiones aplicar a los datos (variables y constantes) del problema algún tipo de operador, para generar nuevos valores y/o analizar el contenido de estos valores con algún tipo de comparación, permitiendo llegar a la solución.

Actividad 1

Solucione el siguiente problema:

¿Cuántas manzanas hay en la caja?

Tenga en cuenta que no está permitido contar una a una cada manzana de la caja.



Aplique los pasos de la guía anterior para lograr la solución del problema.

Referencia 2

Para lograr dar solución al problema anterior usted aplico un operador aritmético, sin embargo, debemos entender que existen 3 categorías de operadores:

- ✓ Aritméticos
- ✓ Lógicos
- ✓ Relacionales

Cada una de estas categorías tienen un propósito y los operadores que componen estas categorías tienen una función específica.

Operadores Aritméticos

Es un símbolo matemático que indica que debe ser llevada a cabo una operación específica sobre dos datos.

Nombre	Operador
Suma	+
Resta	-
Multiplicación	*

Nombre	Operador
División	/
Modulo	Mod

La gran mayoría de los operadores aritméticos son conocidos y manejados, sin embargo, es importante apropiarnos del funcionamiento de los operadores división y modulo.

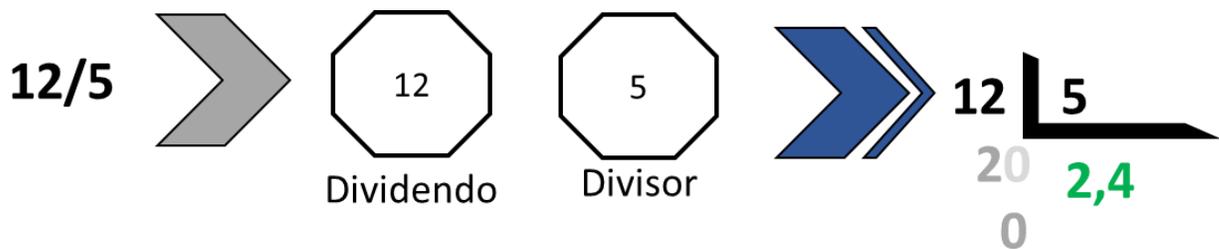
Dividir 16/2.

Para empezar recordaremos las partes de una división.

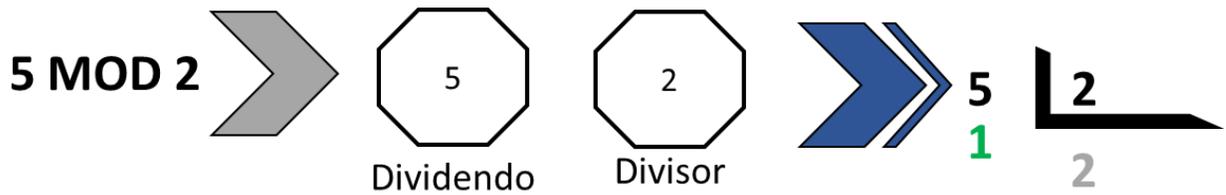


Al desarrollar la división se pueden obtener dos respuestas el cociente y el residuo. Es posible conocer estos dos valores con ayuda de los operadores / y MOD.

Ejemplo División



Ejemplo Modulo



Importante el módulo realiza la división hasta su parte entera.

Reto 1:

Complete la siguiente tabla:

N°	Operación Matemática	Resultado
1	$23 / 5$	
2	$37 \text{ Mod } 5$	
3	$(814/11) \text{ Mod } 2$	
4	$\{[(5*4)/2] \text{ Mod } 3\}$	
5	Cree la operación usando /	16
6	Cree la operación usando Mod	0
7	Cree la operación usando + - * / Mod	4

Operadores Relacionales

Los operadores relacionales son símbolos que se usan para comparar dos valores. Si el resultado de la comparación es correcto la expresión considerada es verdadera, en caso contrario es falsa.

Los operadores de relación son:

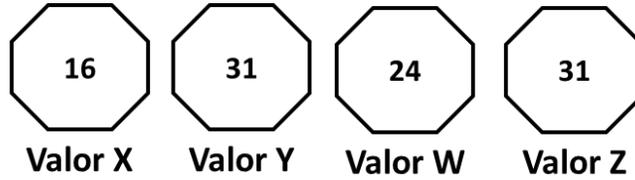
Nombre	Operador	Logica
Mayor que	>	Devuelve verdadero en caso de que el primer operador sea mayor que el segundo, falso en caso contrario.
Menor que	<	Devuelve verdadero en caso de que el primer operador sea menor que el segundo, falso en caso contrario.
Igual que	=	Devuelve verdadero en caso de que los dos operadores sean iguales, falso en caso contrario.

Ejemplo



Reto 2:

Complete la siguiente tabla teniendo en cuenta que:



N°	Operador Relacional	Resultado
1	$X < W$	
2	$Z = Y$	
3	$X > Z$	

Operadores Lógicos

Los operadores lógicos proporcionan un resultado a partir de que se cumpla o no cierta condición.

Los operadores lógicos son:

Operador	Logica
Y	Devuelve verdadero en caso de que los dos operadores sean verdaderos, falso en caso contrario.
O	Devuelve verdadero en caso de que alguno de los dos operadores sea verdadero, falso cuando los dos operadores son falsos.
No	Devuelve el valor negado del operador, si es verdadero devuelve falso y si es falso devuelve verdadero.

Actividad 1

Daniel quiere salir hoy a jugar un partido de Futbol sus padres le darán permiso si arregla su cuarto y lava los platos que están sucios en la cocina.



Analice todas las posibilidades de Daniel para que le den el permiso de ir a jugar

Arregla el Cuarto	Lava los Platos	Arregla el Cuarto Y Lava los Platos
<i>SI</i>	<i>SI</i>	
<i>SI</i>	<i>NO</i>	
<i>NO</i>	<i>SI</i>	
<i>NO</i>	<i>NO</i>	

Reto 3:

Diseñe un problema donde se pueda crear la tabla de posibilidades para el operador lógico "O"

Referencia 2

Ubique en la pestaña de Programas la categoría "Operadores" y observe los diferentes bloques:

Operadores aritméticos:



Los operadores de relacion



Y los operadores lógicos



Referencia 3

Condicionales

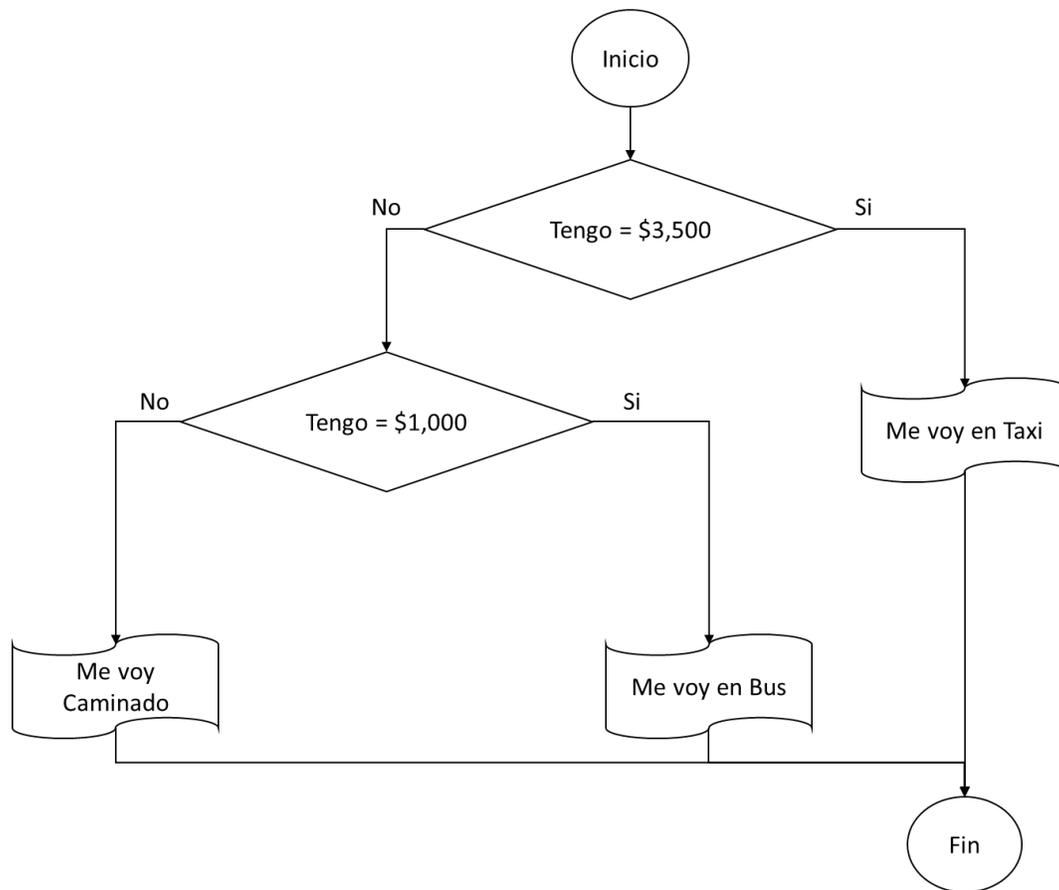
En la vida diaria nos enfrentamos a problemas que para poderlos solucionar debemos analizar la situación y dependiendo de ese análisis se decide que acción realizar.

Analicemos la siguiente situación: Paola quiere ir a su casa y analizando sus posibilidades puede:

- ✓ Irse caminando
- ✓ Tomar un taxi
- ✓ Tomar un bus

Asumiendo que ella quiere llegar a su casa lo mas pronto posible.

- ✓ ¿Cuál sería el orden de prioridad de las opciones anteriores?
- ✓ ¿Pero puede Paola salir y tomar un taxi solo por que es su mejor opción?
- ✓ ¿Después de que Paola organiza sus opciones que debe analizar para determinar cuál escoger?



¿El algoritmo anterior es Finito?

¿Es claro el Inicio y Fin?

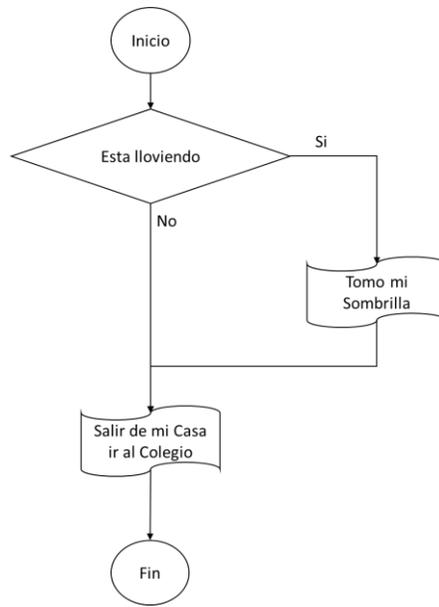
¿La lógica de cada condicional usa el operador correcto?

Tipos de estructuras condicionales:

Condicionales Simples

Las estructuras condicionales simples se les conocen como tomas de decisión en un solo sentido; esto quiere decir que al aplicarla si la condición es verdadera se ejecuta el conjunto de instrucciones y no existe una contraparte a ejecutar si la condición es falsa.

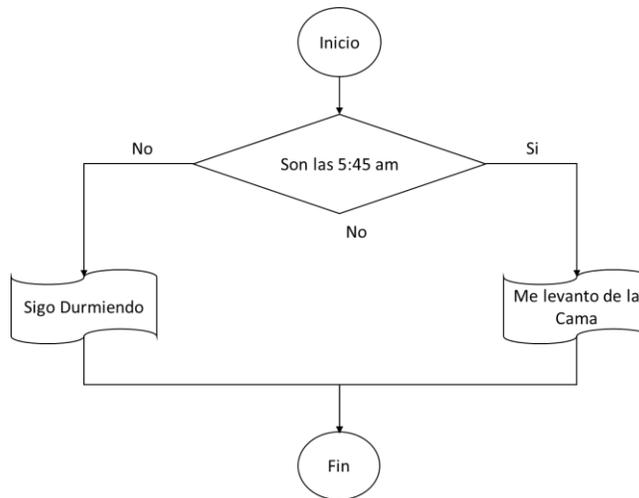
Ejemplo:



Condicionales Dobles

Las estructuras condicionales dobles manejan una condición y dos conjuntos de instrucciones diferentes, donde si la condición es verdadera se ejecuta el conjunto de instrucciones 1 y si la condición es falsa se ejecuta el conjunto de instrucciones 2.

Ejemplo



Condicionales Anidados

Una estructura condicional es anidada cuando existe un condicional y en la rama del No se construye otro condicional y así sucesivamente.

Ver ejemplo de Paola.

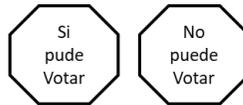
Actividad 2

Solucione el siguiente problema haciendo uso de Scratch.

Julián tiene 22 años y quiere saber si puede votar en las próximas elecciones.

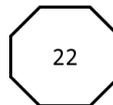
Análisis del Problema

¿Que solución se debe entregar?



Respuesta

¿Qué datos Constantes y Variables conozco?



Edad de Julian

Nada

¿Que está prohibido hacer?

Todo

¿Que está permitido hacer?

Analizar si
Edad > 17

¿Qué Análisis/formulas debo emplear?

1. Almacenar en una variable la Edad "X"
2. Aplicar un Condicional Doble y si es verdadero informar que si puede votar, pero si es falso informar que no puede votar.

¿Qué pasos debo seguir para obtener la solución?

1. Inicie un nuevo Proyecto en Scratch
2. Cree una variable de nombre "X"
3. Inicie con la estructura básica de un programa en Scratch y asigne a la variable "X" la edad de Julián.
4. En la Pestaña Programas ubique la categoría Control



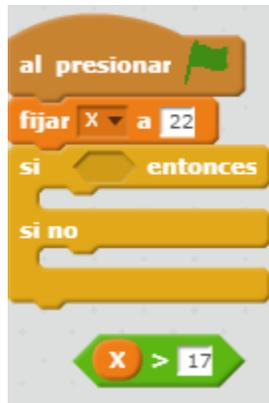
5. Ubique el bloque Si entonces Si No



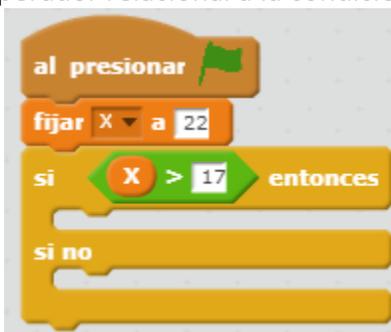
Este condicional tiene doble boca en forma de "C", en la que se pueden insertar otros bloques en su interior. También tiene un espacio que termina en punta donde se puede encajar otros bloques.

Funcionamiento del Si ____ Entonces Si-no: se evalúa la condición, y si verdadera, ejecuta la secuencia de acciones que hay debajo del Si. Si la condición fuese falsa, se ejecuta los bloques o las secuencia de acciones que existan debajo del si no.

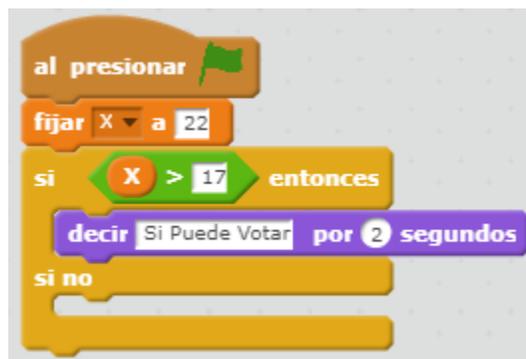
6. En la categoría operadores seleccione el bloque "mayor que" e integre a este la variable X y el numero 17



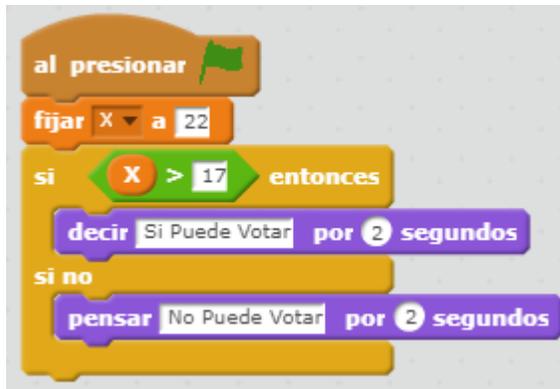
7. Mueva el bloque del operador relacional a la condición del Si



8. Agregue un mensaje al conjunto de instrucciones del "Si" para informar que si puede votar.



9. Agregue otro mensaje esta vez para el conjunto de instrucciones del "Si No" informando que no puede votar.



Cambie la edad de Julián a 14 y observe el funcionamiento.

Reto 4:

Cree un programa en Scratch donde implemente el algoritmo de Paola.

Tip: Los condicionales anidados se pueden forma de la siguiente forma.



Reto 5:

Diseñe un programa en Scratch para indicar a un padre de familia cuanto medicamento debe suministrar a un niño de 4 años que pesa 23 kg. Donde la dosificación según la caja del medicamento es:

Edad del Niño	Suministrar Medicamento
0 a 2 Años	30 Gotas
3 a 5 Años	5 ml
6 años en adelante	3 ml por cada 2Kg de peso

Anexo 4

Guía 04

Recapitulando la guía anterior

En la guía anterior abordamos el tema de los operadores lógicos “Y” “O” y se generaron las tablas de verdad para cada uno de ellos. Complete las siguientes tablas de verdad, determine que operador lógico debe usar cada tabla y determine el resultado:

Hacer Dieta	Hacer Ejercicio	Voy a bajar de Peso si: Hago Dieta ____ Hago Ejercicio
<i>SI</i>	<i>SI</i>	
<i>SI</i>	<i>NO</i>	
<i>NO</i>	<i>SI</i>	
<i>NO</i>	<i>NO</i>	

Tener Licencia de Conducción	Tener carro	Puedo conducir un Carro por mi ciudad si: Tengo Carro ____ Tengo Licencia
<i>SI</i>	<i>SI</i>	
<i>SI</i>	<i>NO</i>	
<i>NO</i>	<i>SI</i>	
<i>NO</i>	<i>NO</i>	

Complete la siguiente regla para los operadores lógicos “Y” “O”. (Puede ajustar los plurales o singulares de las palabras.)

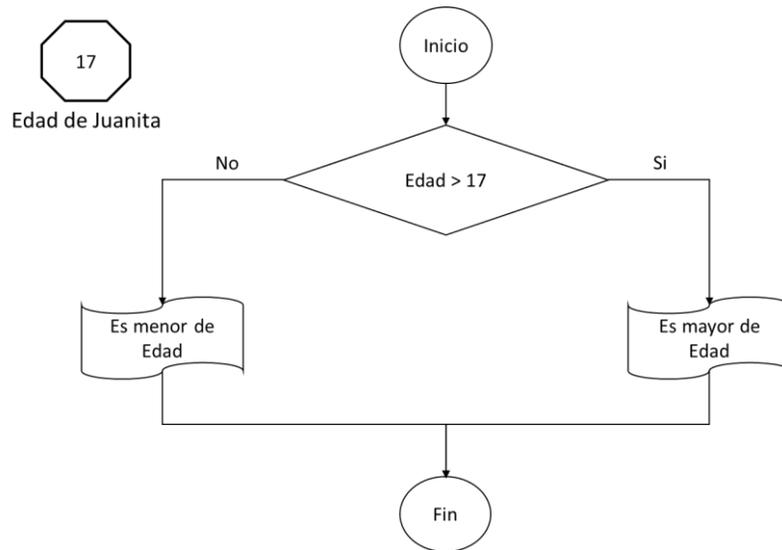
- ✓ Dos proposiciones A y B unidas por un operador lógico ____ se considera verdadero el resultado si _____ proposiciones son verdaderas.
- ✓ Dos proposiciones A y B unidas por un operador lógico ____ se considera verdadero el resultado si _____ proposiciones son verdaderas.

Explique con sus propias palabras que es un condicional y ¿que función tiene?

Referencia 1

Hasta el momento se han realizado ejercicios donde los datos a procesar son constantes, pero no siempre es así...

Por ejemplo: Juanita tiene hoy 17 años si preguntásemos hoy si ella es mayor de edad la respuesta sería “NO”...



Mañana Juanita esta de cumpleaños!! Si ejecutamos el algoritmo anterior mañana cual seria el resultado? Correcto..... el mismo “Es menor de edad”... y si ejecutamos el algoritmo en 20 años cual seria el resultado....

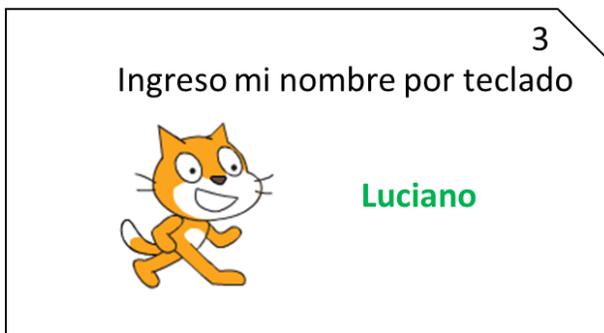
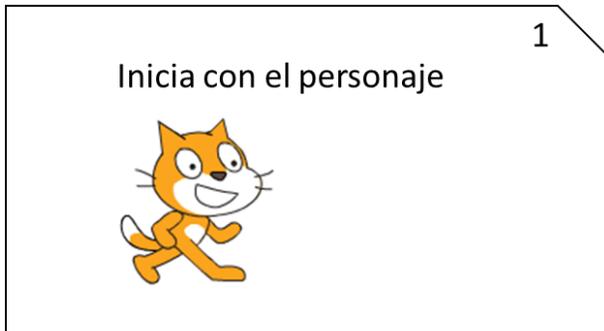
¿Porque ocurre esto?

¿Que debería ocurrir en el algoritmo para que funcione correctamente?

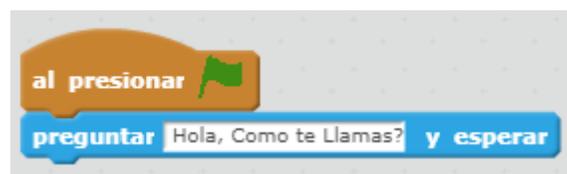
Actividad 1

Desarrollar un programa en Scratch que nos salude por nuestro nombre.

Describamos con Imágenes y palabras como sería el funcionamiento del programa esto se conoce como crear un story board:



10. Inicie un nuevo Proyecto en Scratch
11. Cree una variable de nombre "Nom"
12. Inicie con la estructura básica de un programa en Scratch (Bloque al presionar) *Story Board 1*
13. En la pestaña programas ubique la categoría sensores y seleccione el bloque "Preguntar y esperar" (este bloque es el que nos permite ingresar los valores por teclado), en el espacio hacemos la pregunta "¿Hola, como te llamas?" *Story Board 2 y 3*



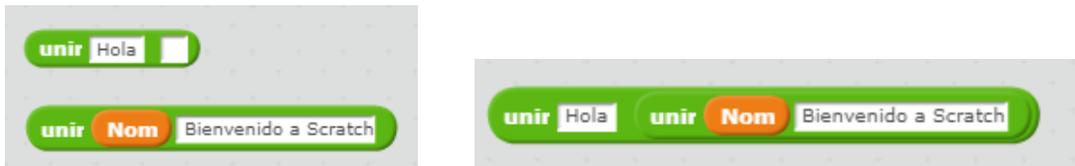
14. Por defecto los valores que se ingresan con este bloque quedan almacenados en la variable interna Respuesta de la categoría sensores.



15. Asigne a su variable Nom el contenido de la variable Respuesta



16. Utilice el bloque unir para crear el texto de saludo final.



17. Agregue el bloque anterior a un bloque Decir. *Story Board 4*



Actividad 2

Realizar un programa en Scratch que solicite un número y posteriormente determinar si es positivo o negativo.

¿Que solución se debe entregar?



¿Qué datos Constantes y Variables conozco?



Numero (Variable dado por teclado)

¿Qué Análisis/formulas debo emplear?

Analizar si
El Numero es > 0

1. Declarar una Variable X
2. Solicitar por teclado el numero
3. Almacenar en la variable "X" el numero
4. Aplicar un Condicional Doble y si es verdadero informar que es positivo, pero si es falso informar que es negativo.

¿Que está prohibido hacer?

Nada

¿Qué pasos debo seguir para obtener la solución?

¿Que está permitido hacer?

Todo

Reto 1:

Contemple las posibilidades de Positivo, Negativo y Neutro (el cero)

- ✓ ¿Que cambio debe realizar?

Reto 2

Realizar un programa para solicitar 3 números diferentes y posteriormente determinar cuál es el número Mayor.

No olvide realizar el análisis del problema



Pista Los operadores Lógicos pueden ser la diferencia entre un código largo o uno corto

Reto 3

Realizar un programa para solicitar un número y posteriormente determinar si es par o impar.

No olvide realizar el análisis del problema



Pista Los números pares son divisibles entre 2...

Es decir, si el número se divide entre 2 y el residuo da 0, es par.

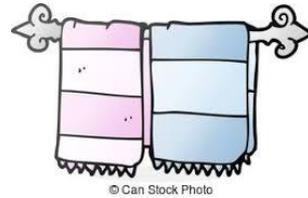
Anexo 5

Guía 05

Recapitulando la guía anterior

Referencia 1

En la vida diaria debemos realizar acciones de forma repetitiva hasta conseguir un objetivo. Por ejemplo pensemos en los pasos para lavarnos las manos. Suponga que el lavamanos tiene una única llave y que el jabón es líquido de dispensador.



Algoritmo

1. Girar la llave a la derecha
2. Mojar las manos
3. Oprimir el dispensador de jabón 3 veces, recibiendo el jabón en la palma de la mano.
4. Frotar las manos mojándolas un poco hasta conseguir abundante espuma.
5. Enjuagar las manos para quitar el Jabón
6. Sacudir las manos
7. Girar la llave a la Izquierda
8. Secar las manos con una Toalla

Analícemos un poco:

- ✓ Que Instrucciones son puntuales y se realizan solo una vez.
- ✓ Que Instrucciones requieren que se repitan hasta lograr un objetivo.
- ✓ Como explicaría con sus propias palabras cada instrucción.
- ✓ Existe diferencia entre la forma de realizar estas instrucciones.

Instrucciones Repetitivas (Ciclos)

Las estructuras que repiten una secuencia de instrucciones un número determinado de veces se denominan ciclos (bucles) y se denomina iteración al hecho de repetir la ejecución de una secuencia de acciones.

En Scratch se encuentran 3 tipos de estructuras repetitivas (ciclos), en la categoría de "Control", entre ellas encontramos

- Repetir.
- Por siempre.
- Repetir hasta que.



Para entender el funcionamiento de estas instrucciones analicemos los siguientes apartados:

Bloque Repetir: trae un espacio en el cual se indica la cantidad de veces que se debe repetir el conjunto de instrucciones, que esta dentro de el. (por defecto trae el número 10).

Ejemplo: Lanzar un balón de baloncesto con 10 intentos, para ver cuantas cestas hace.

Cuando se conoce cuantas repeticiones se van a realizar se usaría un Bloque Repetir, y dentro de este se indicaría el conjunto de instrucciones:



- a) Lanza Balón
- b) Un condicional Si, para validar si se realizó la cesta y si es así contar un punto.

Bloque Por Siempre: Es un ciclo que se activa y se ejecuta de forma infinita, solo se detiene cuando ocurre un suceso especial que ejecuta una instrucción de fin de ciclo.

Ejemplo: Ahorrar para comprar un Celular.

Usted no sabe cuanto tiempo va a durar ahorrando, por esto ejecutara un ciclo por siempre, y su conjunto de instrucciones sera depositar dentro de la alcancia el dinero. Cuando el total ahorrado sea Igual al valor del celular, usted cancelara la ejecucion del ciclo.



- a) Depositar dinero dentro de la Alcantia
- b) Si el dinero almacenado es igual al valor del celular "Finalizar el Ciclo".

Bloque Por Repetir hasta que : Es similar al bloque Repetir con la diferencia que no controla la cantidad de repeticiones que debe realizar, en su lugar maneja una condición lógica, que mientras sea verdadera el ciclo se ejecutara, pero cuando se vuelva falsa el ciclo terminara.

Ejemplo: Comer hasta estar satisfecho

Usted se encuentra en una plaza de comidas y puede comer todo lo que quiera, pero usted sabe cuánto va a comer.... Su condición seria "Tengo Hambre" si esto es cierto ejecuta el conjunto de instrucciones. Cuando sea falso termina el ciclo.



Condición ¿Tengo Hambre?

- a) Buscar un Alimento
- b) Comer Alimento

Actividad 1

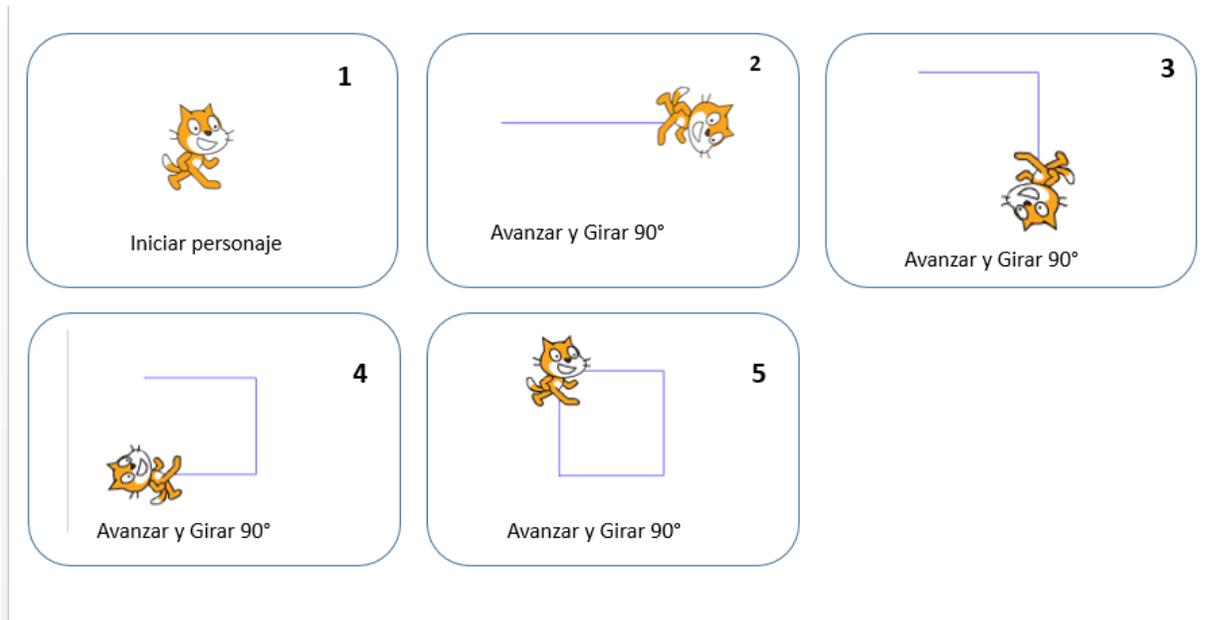
En el problema de lavarse las manos con los ciclos que identifico, ¿cuales bloques serian los correctos para implementar esas rutinas?

Actividad 2

Usando Scratch dibujar un cuadrado, utilizando los bloques de LAPIZ.

Nota: Debe usar el ciclo repetir

Describamos con Imágenes y palabras como sería el funcionamiento del programa esto se conoce como crear un story board:



18. Inicie un nuevo Proyecto en Scratch
19. Inicie con la estructura básica de un programa en Scratch (Bloque al presionar)
20. En la pestaña programas ubique la categoría "Control" y seleccione el bloque Repetir y en el espacio colocamos 4 que sería cada lado del cuadrado.



21. Ahora dentro del ciclo vamos a incluir una nueva herramienta la cual hace el rol de lápiz. Este se encuentra en la pestaña de lápiz y seleccionamos el bloque bajar lápiz el cual hara el trazo en la pantalla, quedando de esta manera.

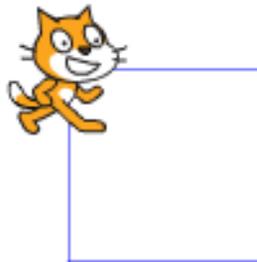


22. Por ultimo usaremos dos bloques ya usados con anterioridad el bloque mover__pasos y girar__grados, colocando los valores 150 y 90 respectivamente e integrándolo dentro el ciclo así(para observa mejor el movimiento podemos usar el bloque “esperar 1 segundo” en la pestaña de control)

Antes de Ejecutar el programa en Scratch tomese su tiempo para imaginar el funcionamiento,



23. Ejecute el programa y observe el resultado.



24. Observemos que sin el uso del ciclo tendríamos que repetir la operación 4 veces algo por el estilo.



25. Que puede concluir comparando ambos codigos.

Actividad 3

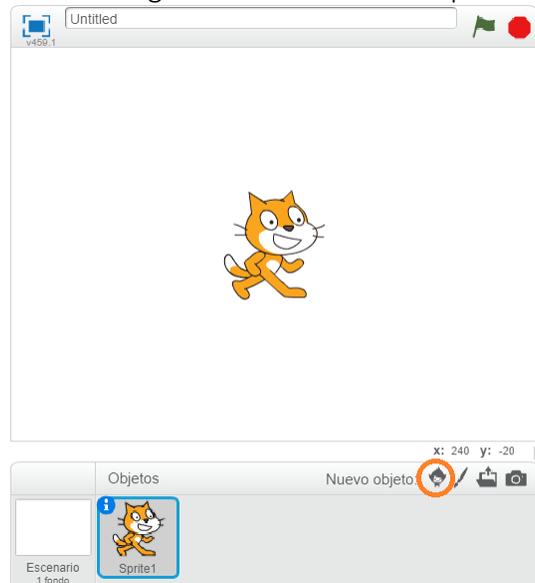
Realizar un programa en Scratch que visualice un fantasma que se desplace cierto tiempo, desaparezca y luego aparezca.

Para este ejercicio no haremos el análisis normal que realizamos debido a que no nos generara ninguna respuesta como tal sino una reacción.

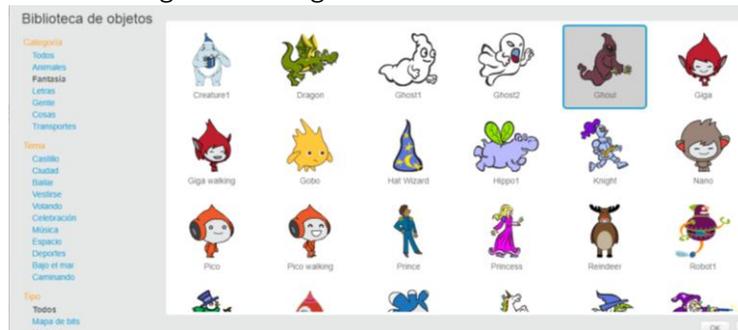
- ✓ ¿Qué necesitamos?
Cambiar el icono de al gato por el de un fantasma y el fondo de pantalla para animar más el entorno.
- ✓ ¿Qué tiene que hacer el fantasma?
Caminar, aparecer y desaparecer.
- ✓ ¿Cómo hacerlo?
Implementando un ciclo para hacer el recorrido.

Desarrollo:

1. Iniciamos el programa
2. Nos ubicamos en la parte superior del explorador de objetos (círculo rojo) esta opción nos permite escoger diferentes estilo de personajes.

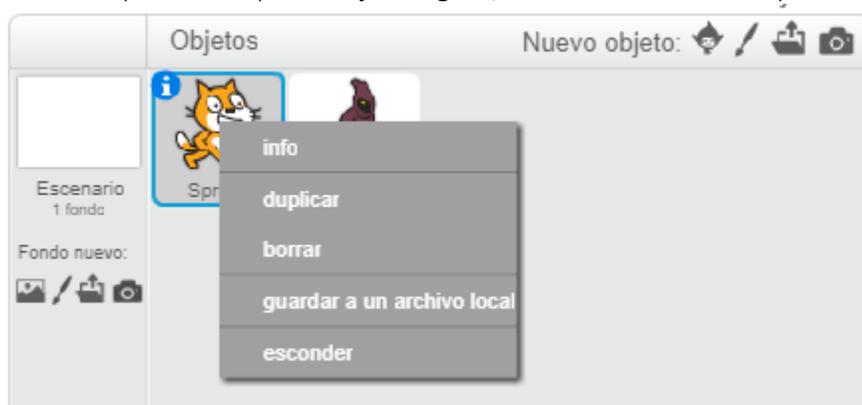


3. Luego nos saldrá la siguiente imagen

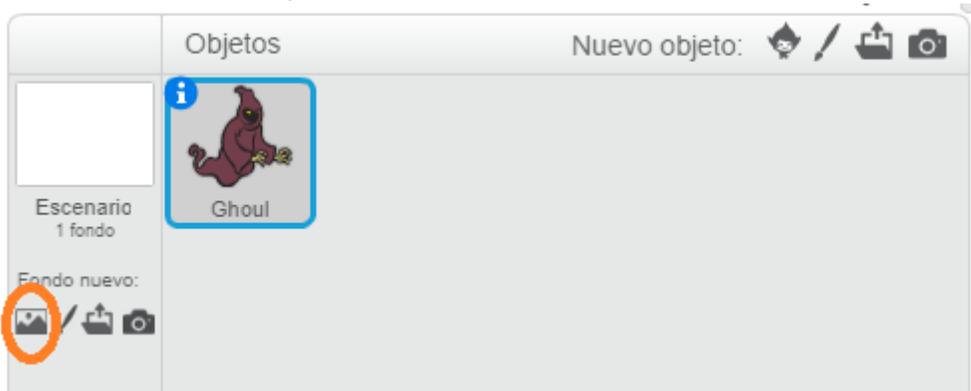


Seleccionamos la categoría fantasía y escogemos el fantasma que dese.

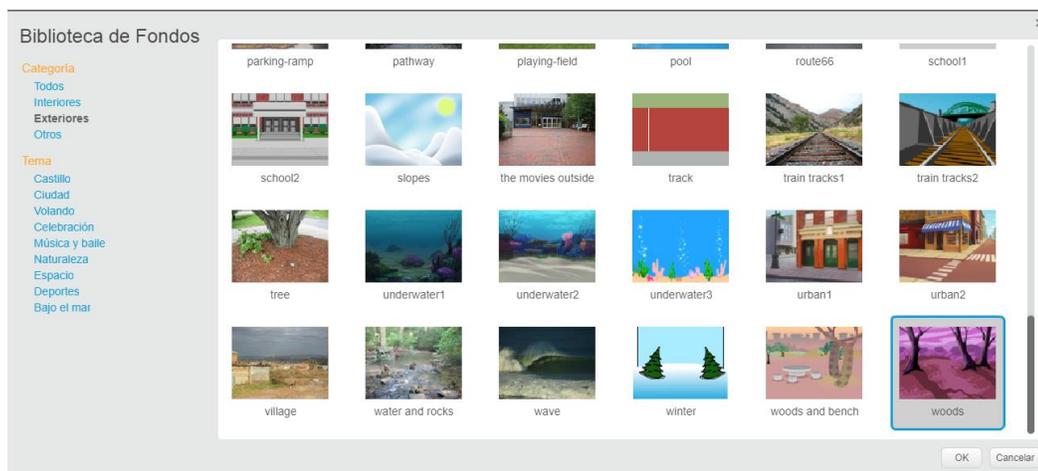
4. Para desaparecer el personaje del gato, clic derecho sobre el y seleccionar Borrar



5. Ahora cambiaremos el fondo de pantalla. Damos clic en el icono donde se encuentra el círculo rojo

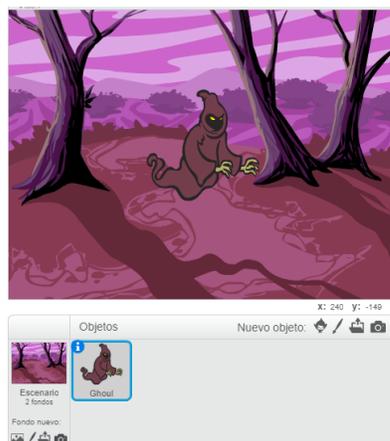


6. Aparecerá la siguiente ventana



Nos ubicamos en la categoría de exteriores y buscamos la opción Woods como aparece en pantalla.

Para validar que si quedo agregado observamos el fondo de pantalla donde esta el fantasma saliendo esto.

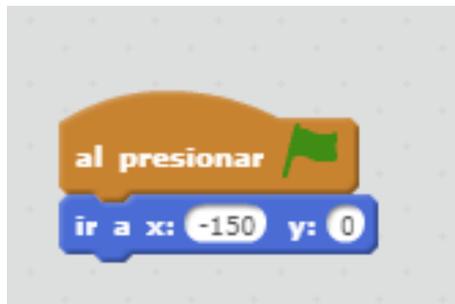


7. Ya teniendo el entorno grafico pasaremos a la estructura funcional, regresamos a la pestaña Programas y teniendo seleccionado el personaje fantasma, añadiremos para este ejercicio dos entradas de eventos el cual será “al presionar”
8. El primero será para ubicar el inicio de posición de nuestro fantasma y el otro llevara las instrucciones de aparecer y desaparecer.

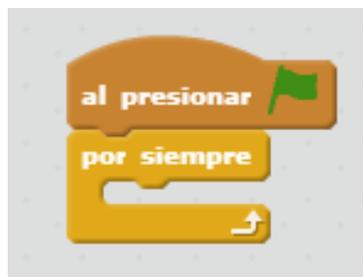


Ahora uno de los “al presionar “ llevara solamente el bloque de “ir a X: Y”: (este bloque se encuentra en la pestaña de movimiento)

X tendrá el valor de -150
Y el valor de 0



9. En el otro bloque “Al presionar” lo usaremos con un ciclo “por siempre” que cumplirá la función de repetir el movimiento del fantasma



10. Ahora dentro del bloque por siempre vamos a colocar un condicional de si _ entonces.



11. Ahora usaremos algo que no hemos visto, el realizar una acción oprimiendo una tecla

Para esto contamos con un bloque que se llama “tecla presionada ” el cual nos deja seleccionar alguna de las teclas del teclado, para realizar acciones. este bloque se encuentra en la sección de sensores



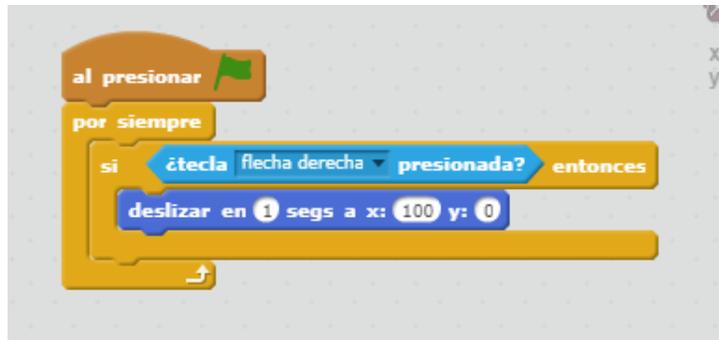
12. Por defecto usa la tecla espacio nosotros usaremos la tecla “flecha derecha ” y la incluiremos como condicional del si entonces.



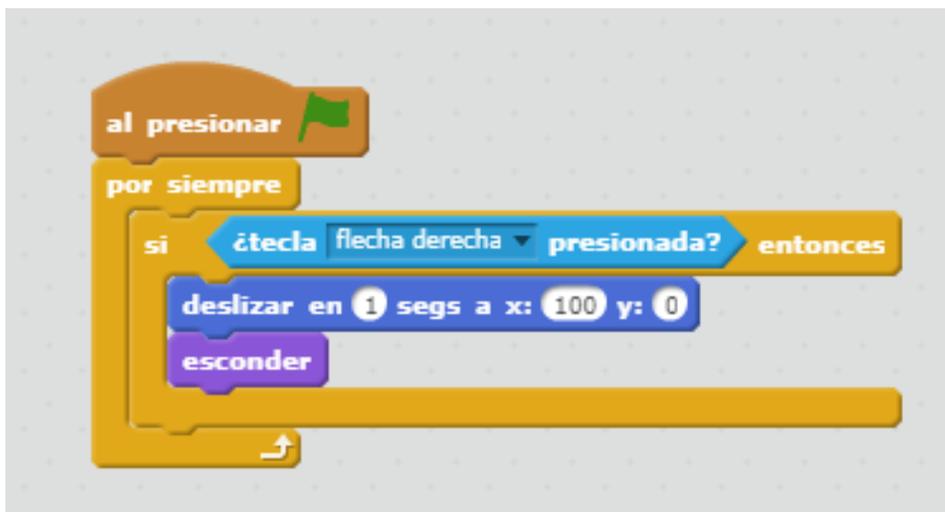
13. A continuación incluiremos un bloque de la sección de movimiento llamado “deslizar en ”



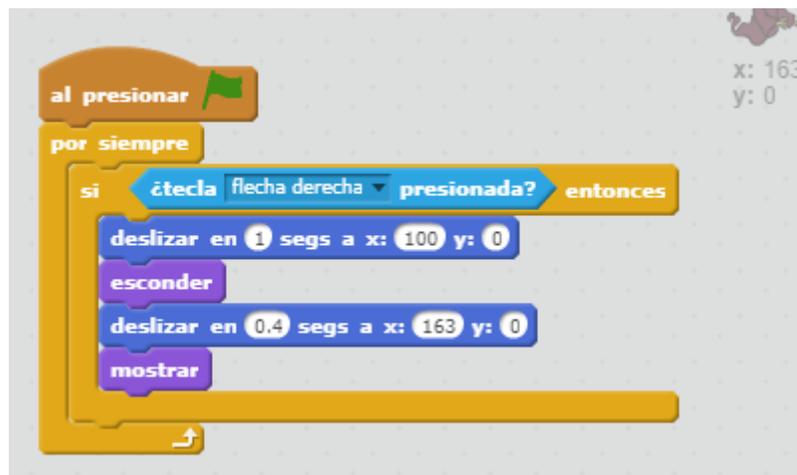
Configure los valores de la imagen en el bloque y agréguela dentro del si entonces



14. Seguido de esto usaremos de la pestaña apariencia el bloque “esconder” el cual nos ocultara el personaje y lo conectamos al anterior bloque



15. Para que aparezca usaremos los últimos dos bloque pero le cambiaremos los valores del deslizar en a los valores que aparecen en la imagen, y en vez de colocar el bloque esconder usamos el bloque “mostrar”



Reto 1:

Modifique el recorrido del fantasma para que se deslice de arriba a bajo y de izquierda a derecha.

Reto 2

Que le agregaría para que cuando se pulse la flecha Izquierda inicie el fantasma de nuevo en la posición inicial.

Anexo 6

Guía 06

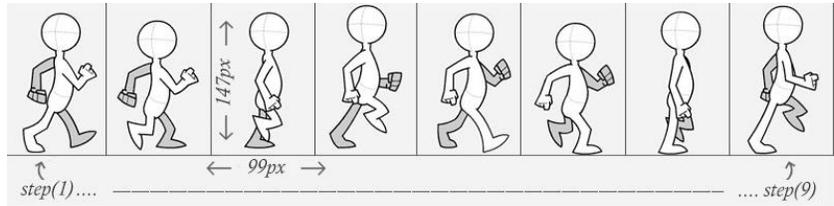
Para entrar en contexto

- ✓ ¿Qué es una animación?
- ✓ ¿Cómo explicaría el funcionamiento de una caricatura?
- ✓ ¿Alguna vez ha dibujado en la esquina de su cuaderno una imagen con cambios sutiles en cada hoja, para después pasarlas rápidamente?

¿Qué es un Flip Book o Foliscopio?

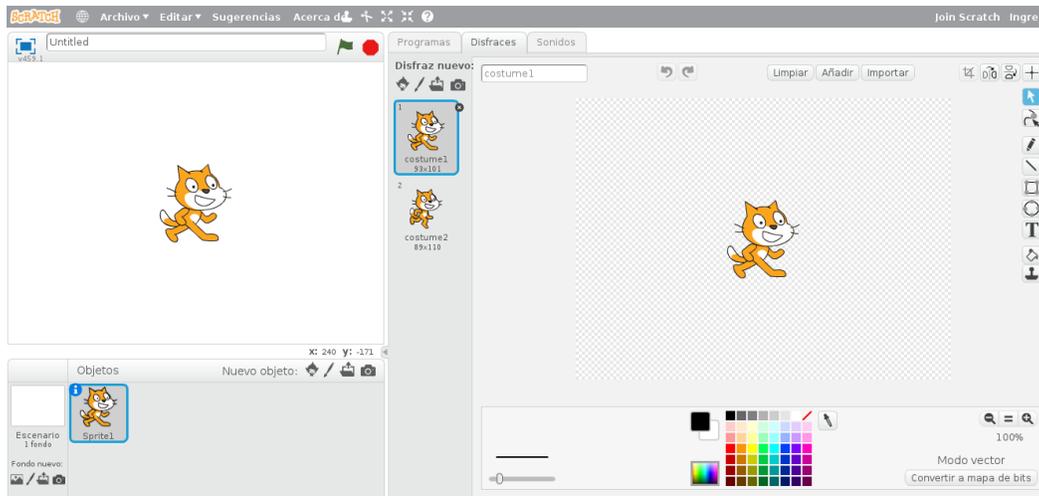
Es un libro que contiene una serie de imágenes que varían gradualmente de una página a la siguiente, para que, cuando las páginas se pasan rápidamente, las imágenes parezcan animarse simulando un movimiento u otro cambio. Son ilustrados usualmente por niños, pero pueden estar también orientados a adultos y emplear una serie de animaciones.

Los folioscopios son esencialmente una forma primitiva de animación.



Actividad 1

Scratch permite realizar a los personajes cambios de apariencia, esto se conoce como cambios de Disfraz. El personaje Gato que viene al iniciar un proyecto, tiene dos disfraces, observe la pestaña Disfraces, existen 2: Costumer1 y Costumer2



Analice el siguiente bloque de código:



Interprete con sus propias palabras que hace el código:

Realice un proyecto en Scratch con el código anterior y observe el funcionamiento.

Reto 1:

Implemente en la actividad anterior que el Personaje Gato solo camine cuando se pulse la tecla Flecha Derecha del teclado.

Nota: En Sensores esta un bloque que podría ser un buen recurso para solucionar el reto.



Actividad 2

En esta actividad haremos la inclusión de un personaje nuevo (Fuera de los predeterminados), dándole movimiento con las teclas de flechas del teclado.

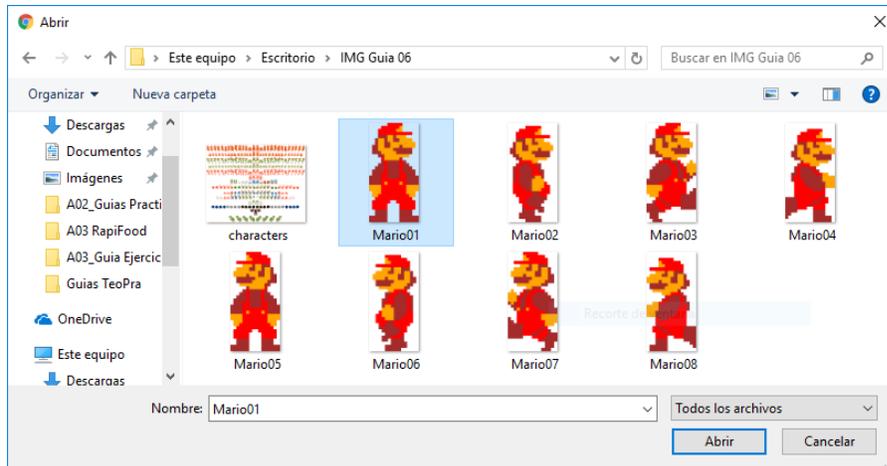
1. Cree una carpeta que contenga las imágenes de Mario Bros y ubíquela en el escritorio.



2. Inicie un nuevo proyecto en Scratch.
3. Para crear el personaje de Mario, debe ir a Nuevo Objeto y hacer clic en el icono de la carpeta.



4. Seleccione la primera imagen de "Mario01"



5. Una vez creado el nuevo personaje, elimine el personaje de Gato del proyecto haciendo clic derecho sobre el y seleccionado borrar.



6. Una vez creado el personaje se deben cargar todos los cambios de disfraz de mario (Mario02 a Mario04)

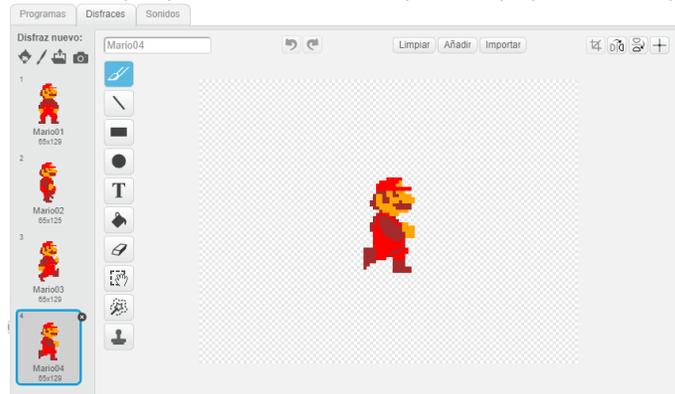
- a. Ubique la pestaña de Disfraces y clic en el icono de la carpeta “Cargar Disfraz desde archivo”



- b. Seleccione los archivos (Mario02 a Mario04)

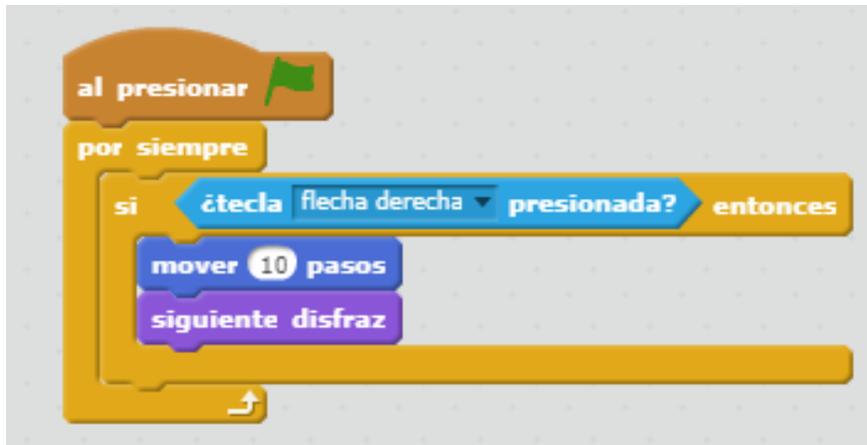


- c. Observe que para nuestro nuevo personaje ya están disponibles 4 disfraces.



7. Es tiempo de programar a Mario para que camine a la derecha al pulsar la flecha del teclado.

Una sugencia de codigo puede ser:



Pero..... tiene algunos errores el movimiento

¿Cuales errores detecta?

¿Que le mejoraría?

Reto 2:

Solucione los errores y las mejoras que detecto.

Reto 3:

Implemente que Mario pueda caminar para la izquierda (Utilice carpeta de imágenes 2)

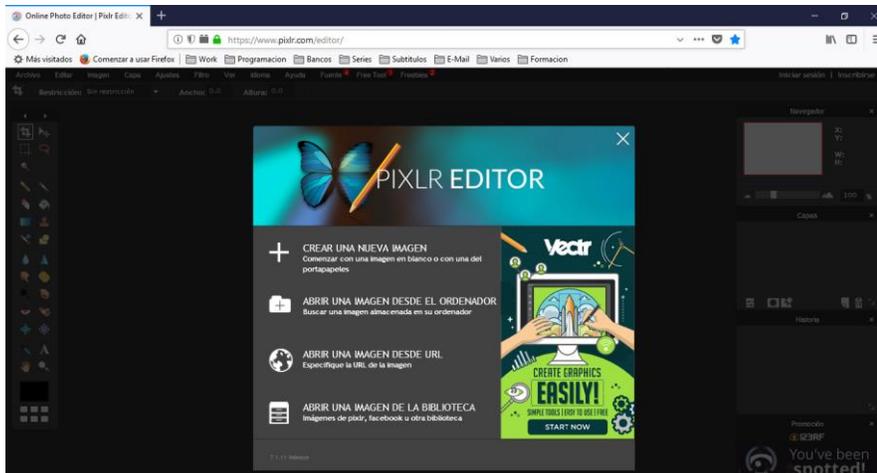
Reto 4:

Implemente que Mario se pueda agachar y que pueda saltar.

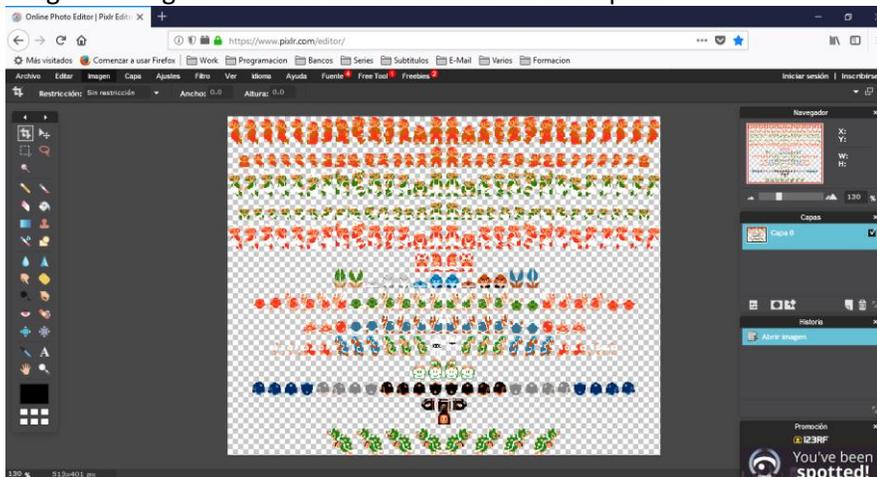
Debe crear las imágenes sacándolas del Sprite "characters" y utilizando alguna herramienta de edición de imágenes.

Nota: Una buena opción para la edición de imágenes es <https://www.pixlr.com/editor/>

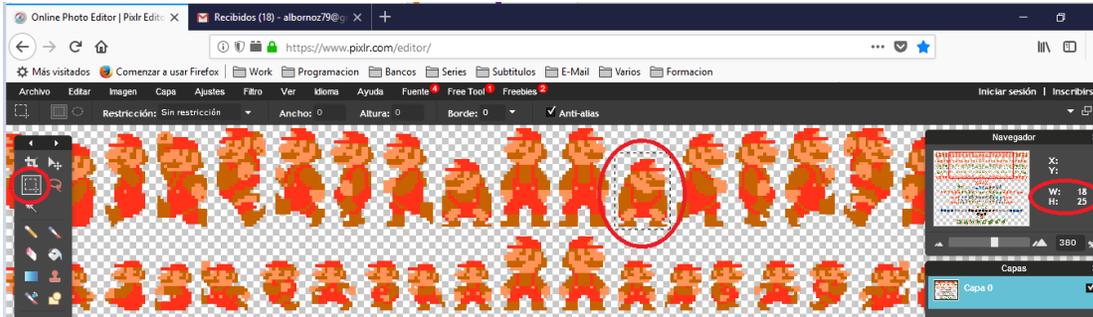
1. Iniciar el PIXLR EDITOR



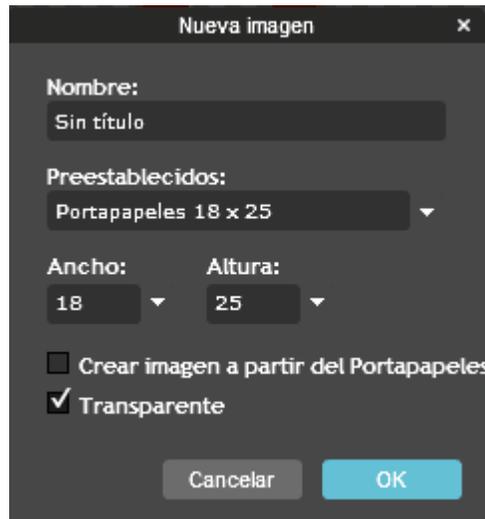
2. Cargar la imagen haciendo clic en el icono de la carpeta



3. Pulsando **Ctrl +** realiza ZOOM de la imagen ubique a Mario agachado y con la herramienta de Marco seleccione a Mario agachado. Peste atención en la parte derecha al tamaño **W** y **H**



4. Realice **Ctrl C** y cree una nueva imagen. Archivo -> Nueva Imagen



Observe que se configuran las propiedades para el tamaño de la imagen seleccionada, seleccione Transparente. Debe hacer coincidir el tamaño de sus otros marios con este nuevo que está creando. (65x129)

5. Realice **Ctrl V**
6. Guarde la imagen.