



UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

Enseñanza y aprendizaje en línea con base en
WeScheme

TESIS
PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTA:

LUIS GONZALO FLORES HERRERA

DIRECTOR DE TESIS:

Mtro. JUAN JOSÉ CARREÓN GRANADOS



Cd. Universitaria, D.F. 2014



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Índice

CAPÍTULO 1. Introducción

1.1 Descripción del trabajo	3
1.2 Objetivo, problema, e hipótesis.....	4

CAPÍTULO 2. Identificación y evaluación de recursos web de aprendizaje

2.1 La educación en línea en la Facultad de Ingeniería.....	5
2.1.1 División de Ciencias Básicas de la Facultad de Ingeniería.....	5
2.1.2 Coloquio 2014	7
2.2 La educación en línea en la Universidad Nacional Autónoma de México.	8
2.2.1 Página principal de la Universidad Nacional Autónoma de México.....	8
2.2.2 Colegio de Ciencias y Humanidades (CCH).	10
2.2.3 Proyecto Arquímedes	12
2.3 La educación en línea en México.	13
2.3.1 Universidad Abierta y a Distancia de México (UnADM, 2014)	13
2.3.2 Universidad Nacional Autónoma de México y Tecnológico de Monterrey.....	14
2.3.3 Centro de Recursos para la Enseñanza y el Aprendizaje.....	15
2.4 La educación en línea a nivel internacional	17
2.4.1 Wikipedia	18
2.4.2 YouTube.....	21
2.4.3 <i>KhanAcademy</i>	23
2.4.4 <i>Coursera</i>	26
2.4.5 <i>Bootstrap</i>	29

CAPÍTULO 3. WeScheme y el aprendizaje de Matemáticas

3.1 Talleres para la enseñanza.....	32
3.2 Interactuando con el entorno de programación.	41
3.3 Diseño de Programas.....	48
3.4 Programación.....	52

3.4.1	Programación <i>batch</i>	52
3.4.2	Programación interactiva	57

CAPITULO 4 Desarrollo e integración de recursos

4.1	Metodología de integración	61
4.2	Ejemplos de integración	61

Conclusión

Bibliografía

CAPÍTULO 1

Introducción

1.1 Descripción del trabajo

El objetivo de esta tesis es explorar opciones de solución a problemas de aprendizaje de antecedentes de matemáticas de estudiantes de primer ingreso de la Facultad de Ingeniería¹ (F.I, UNAM, 2014), lo que a su vez podría poner al alcance de la comunidad universitaria y de todos los interesados un saber sobre estos temas.

De ahí que el capítulo 2 se refiera a diferentes recursos web, incluyendo *MOOC*² (Massive Open Online Courses, 2014) más conocidos. Detallando un poco sus diferencias, métodos de enseñanza, ligas a universidades, características de las plataformas educativas que emplean, así como su relevancia. Además, se mencionan la existencia de sitios importantes de apoyo gratuito a la educación. Se describe la importancia de *YouTube*³ (Youtube, 2014), pues no solo se usa para el entretenimiento, sino también en la educación, siendo de gran apoyo a estudiantes, ya que los cursos y lecciones (de todos tamaños) que existen ahí se pueden consultar, inclusive desde la comodidad del hogar, las veces que sea necesario.

Se consideran también los diferentes y significativos recursos que tiene la Facultad de Ingeniería, FI, para resolver problemas de antecedentes matemáticos de los alumnos de la institución, lo que tiene trascendencia al ser aprovechable por otros alumnos, por ejemplo los de educación media superior; quienes pueden consultar lo que requieren para las materias que cursan, como las que llevarían eventualmente en la Facultad.

Igualmente, se consideran las formas en que se proporcionan conocimientos en la red; cuáles sus ventajas y desventajas en sus formas diversas de proporcionarse; principalmente por las universidades mexicanas más reconocidas, mencionando a las que participan en plataformas educativas extranjeras.

¹ Facultad de Ingeniería de la UNAM, <http://www.ingenieria.unam.mx/> (consultado el 10 de septiembre del 2014)

² MOOC.ES, <http://www.mooc.es/que-es-un-mooc/> (consultado el 10 de septiembre del 2014)

³ YouTube, <https://www.youtube.com/> (consultado 10 de septiembre del 2014)

Así como aplicaciones web, *WeScheme*, que mediante programación distribuida ayudan al aprendizaje de las matemáticas de nivel medio (Geometría, Álgebra y Trigonometría).

1.2 Objetivo, problema, e hipótesis

Objetivo: facilitar mediante recursos y una aplicación web el aprendizaje y el repaso de temas de matemáticas, como Álgebra, Geometría Euclidiana y Trigonometría, a estudiantes de la FI, así como de secundaria y bachillerato, entre otros.

Problema: un grupo significativo de estudiantes que ingresan a la FI carecen de los antecedentes necesarios para cursar asignaturas del primer semestre relacionadas con matemáticas, a pesar de existir recursos a nivel facultad, universidad, país, e internacional, que buscan remediar esta situación.

Hipótesis: identificar, evaluar, desarrollar y organizar esos recursos, en particular los existentes en la web, contribuye a facilitar que los estudiantes aprendan y repasen temas de antecedentes de matemáticas que requieren en su desarrollo académico en la FI.

CAPÍTULO 2

Identificación y evaluación de recursos web de aprendizaje

2.1 La educación en línea en la Facultad de Ingeniería

2.1.1 División de Ciencias Básicas de la Facultad de Ingeniería

La Facultad de Ingeniería cuenta con una página web de ayuda para aprender y repasar antecedentes de matemáticas, donde se dispone de documentos en *PDF* que facilitan el aprendizaje de materias en esa área impartidas por la Facultad, esos documentos se encuentran ubicados en la página web de la División de Ciencias Básicas⁴ (DCB, UNAM, 2014), dando clic en la opción de *Coordinaciones Académicas*⁵ (Coordinaciones Académicas, 2014) y después en el botón *Matemáticas*⁶ (DCB, Matemáticas, 2014) encontramos una lista de materias, como se muestra a continuación.

Coordinaciones académicas

► LA DIVISIÓN

► COMUNIDAD ACADÉMICA

► COORDINACIONES ACADÉMICAS

- MATEMÁTICAS
- FÍSICA Y QUÍMICA
- CIENCIAS APLICADAS

► PUBLICACIONES

► SITIOS DE INTERÉS

MATEMÁTICAS

- Álgebra
- Álgebra lineal
- Cálculo diferencial
- Cálculo integral
- Cálculo vectorial
- Geometría analítica

CÁPSULAS TEÓRICAS Y EVALUACIÓN FORMATIVA DE ANTECEDENTES DE BACHILLERATO

EVALUACIÓN DE ANTECEDENTES DE BACHILLERATO

En donde, por ejemplo, activando la liga *Álgebra*⁷ (DCB, Álgebra, 2014) y al ubicar el puntero sobre *Antecedentes de bachillerato* surgen dos opciones *Cápsulas teóricas y evaluación formativa*⁸ (DCB, 2014) y *Evaluación de antecedentes* (DCB, 2014), accionando la primera y dando clic en *Álgebra*⁹

⁴ División de Ciencias Básicas, FI, UNAM <http://dcb.fi-c.unam.mx/> (consultado el 10 de septiembre del 2014)

⁵ División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/> (consultado el 10 de septiembre del 2014)

⁶ División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/> (consultado el 10 de septiembre del 2014)

⁷ División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/Algebra/> (consultado el 10 de septiembre del 2014)

⁸ División de Ciencias Básicas FI, UNAM, <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/index.html> (consultado el 10 de septiembre del 2014)

⁹ División de Ciencias Básicas, Facultad de Ingeniería, <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/lgebra.html> (consultado el 10 de septiembre del 2014)

(DCB, 2014) contamos con opciones que llevan a páginas de evaluación y a documentos de ayuda acerca de temas de álgebra como se muestra en la imagen siguiente.



Descripción del direccionamiento a la opción *Álgebra*

Si damos clic, por ejemplo, en *Exponentes* y después en *Operaciones algebraicas* entramos a una página de evaluación de opción múltiple que según las respuestas proporcionadas las califica de correctas o en caso contrario recomienda revisar temas específicos a fin de poder contestar correctamente, como se muestra en las siguientes imágenes.

? Operaciones algebraicas

1.- Después de reducir la expresión $(-3x - (x+2)(x-4)) + (x+3)(x-1)$ se obtiene

- $x^2 - 3x - 11$
- $x + 5$
- $- 3x - 11$
- $- 3x + 5$

Correcto

? Operaciones algebraicas

1.- Después de reducir la expresión $(-3x - (x+2)(x-4)) + (x+3)(x-1)$ se obtiene

- $x^2 - 3x - 11$
- $x + 5$
- $- 3x - 11$
- $- 3x + 5$

Revisar la multiplicación y simplificación de expresiones algebraicas.

Desafortunadamente una vez que se da la recomendación correspondiente, por ejemplo, “**Revisar la multiplicación y simplificación de expresiones algebraicas**”, no es tan fácil encontrar el recurso correspondiente que permita hacer esa revisión.

Lo cual es de lamentar, pues en general el contenido de la Coordinación Académica de Matemáticas de la División de Ciencias Básicas de la FI es sumamente valioso, pero si se depurara un poco desde la perspectiva de tecnología web sería aún más útil.

2.1.2 Coloquio 2014

Otro ejemplo de recurso valioso en la web de la División de Ciencias Básicas¹⁰, es el representado por el botón *Memoria de eventos*¹¹, en donde existen ligas a recursos significativos, entre las que destaca la del *SEGUNDO COLOQUIO: “COMPARTIENDO EXPERIENCIAS DE ENSEÑANZA BASADAS EN TIC”*¹², en donde no sólo están disponibles la convocatoria, el programa, las ponencias, del segundo coloquio, sino también los del primer coloquio, además de la cuenta twitter *@ticdcbfi*, así como a sitios relacionados.



Mediante este recurso los profesores de la Facultad de Ingeniería comparten ideas sobre conocimientos y experiencias relacionados con las Tecnologías de la Información y la Comunicación (TIC) a lo largo de años recientes, en donde dan su punto de vista acerca de las oportunidades o

¹⁰ División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/> (consultado el 10 de septiembre del 2014)

¹¹ División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/Eventos/> (consultado el 10 de septiembre del 2014)

¹² División de Ciencias Básicas, FI, UNAM, <http://dcb.fi-c.unam.mx/cerafin/ColoquioTIC/> (consultado el 10 de septiembre del 2014)

dificultades que encuentran al emplear TIC en actividades docentes, además de debatir ventajas y desventajas que este uso les ha aportado para el aprendizaje de sus estudiantes.

Ahí también en la opción “Ponencias”¹³, parte del menú desplegable de *Segundo coloquio*, se consulta los resúmenes de las ponencias realizadas, donde se muestran temas referidos a la educación en línea y las matemáticas.

M.C.Q. Alfredo Velásquez Márquez	Videos: Parte I / Parte II
Uso de winplot como apoyo para impartir algunos temas de Geometría Analítica	Resumen ▼
El programa para graficar winplot, se ha empleado con éxito para apoyar la comprensión del tema Curvas en el Plano Polar de la asignatura Geometría Analítica; así también, se han obtenido resultados satisfactorios en la comprensión de algunos subtemas del tema de Superficies de la misma asignatura.	
[Ver extenso]	
Ing. Gustavo Rocha Beltrán	Videos: Parte I
Examen parcial de casa relativo a Funciones, primer tema del curso de Cálculo Diferencial	Resumen ▼
La evaluación del aprendizaje ha sido quizá el aspecto menos satisfactorio de mi labor docente, ya que en muchas ocasiones la he considerado sesgada, insuficiente, inconsistente o ineficiente. El propósito de realizar un examen parcial sobre un tema en particular, a manera de prueba piloto, ha sido la de procurar una vinculación más estrecha entre el vastísimo contenido de los temas, las fuentes selectas de información en libros y en la Red, los recursos didácticos adicionales, que ofrezco a mis alumnos en presentaciones, hojas de cálculo y mapas mentales, y un banco de reactivos también vasto y acorde a todo lo anterior. Presento la experiencia de haber elaborado, aplicado y calificado un examen que fue distinto para cada alumno, y doy las justificaciones para continuar con este proyecto.	

Desafortunadamente no existen suficientes relaciones entre dicho material y el mencionado en 2.1.

2.2 La educación en línea en la Universidad Nacional Autónoma de México.

2.2.1 Página principal de la Universidad Nacional Autónoma de México.

Los recursos de matemáticas en la *UNAM* (Universidad Nacional Autónoma de México, 2014)¹⁴ son extensos por los diferentes niveles de estudios que abarca la institución. El sitio oficial de la *UNAM* cuenta con un apartado llamado *Estudiantes* donde se localiza una opción llamada *Recursos*

¹³ División de Ciencias Básicas, FI, UNAM, *Ponencias*, <http://dcb.fi-c.unam.mx/cerafin/ColoquioTIC/> (consultado el 10 de septiembre del 2014)

¹⁴ Dr. Felipe Bracho Carpizo, *Universidad Nacional Autónoma de México*, <http://www.unam.mx/> (consultado el 10 de septiembre del 2014)

Educativos, la que se enlaza con *Recursos Educativos para Todos*¹⁵ (Recursos Educativos para Todos, 2014) donde existen diferentes sitios enfocados a la enseñanza de matemáticas tales como: *Math Media*¹⁶, *Matemáticas Lecciones Interactivas*¹⁷ y *PUEMAC Matemáticas*¹⁸, cubriendo también otras materias. Como se muestra en las imágenes siguientes.



Recursos Educativos para Todos. UNAM



Sitios web de apoyo para el aprendizaje

En ésta última imagen se muestran algunos enlaces a sitios relacionados con las matemáticas para visualizar diferentes recursos educativos, lo que enriquece la obtención de conocimientos con el uso

¹⁵ CUAED, *Recursos Educativos para Todos*, <http://suayed.unam.mx/recursos/> (consultado el 10 de septiembre del 2014)

¹⁶ CUAED, *MATH MEDIA*, http://cuaed.unam.mx/math_media/ (consultado el 10 de septiembre del 2014)

¹⁷ CUAED, *Matemáticas Lecciones Interactivas*, <http://www.cuaed.unam.mx/matematicas/> (consultado el 10 de septiembre del 2014)

¹⁸ PUEMAC Matemáticas, <http://puemac.matem.unam.mx/> (consultado el 10 de septiembre de 2014)

de las tecnologías webs orientadas a la enseñanza.

En la misma página *Recursos y Servicios Educativos* buscamos dentro del menú la opción *Conocimientos Fundamentales para la Enseñanza Media Superior*¹⁹ (Colección Conocimientos Fundamentales, 2014), la que direcciona a un menú de materias que cursan los estudiantes en el Bachillerato y a información relacionada con cada materia, desafortunadamente conduce finalmente a recursos en CD y bibliográficos poco accesibles.



2.2.2 Colegio de Ciencias y Humanidades (CCH).

El sitio web del Colegio de Ciencias y Humanidades²⁰ (CCH, 2014), en la opción *Portal Académico* se direcciona a la página *Portal Académico del Colegio de Ciencias y Humanidades*²¹, en la misma se pasa el cursor por el botón *Enseñar* donde se muestra una lista de recursos de ayuda, uno de los recursos es el *Banco de Estrategias*²² el cual se direcciona a una página donde se tiene una lista de materiales didácticos de temas de estudio donde se elige el tema *Matemáticas* y nos direcciona a otra página *Área de Matemáticas* y se elige la opción *Matemáticas* donde se tiene un menú *Matemáticas 2*, *Matemáticas 3* y *Matemáticas 4*, si damos clic en alguna de las tres opciones nos

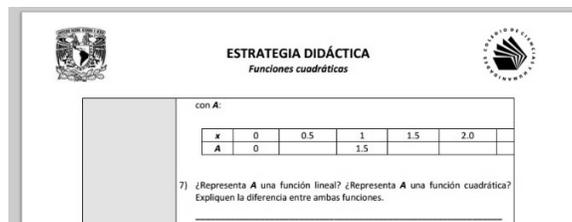
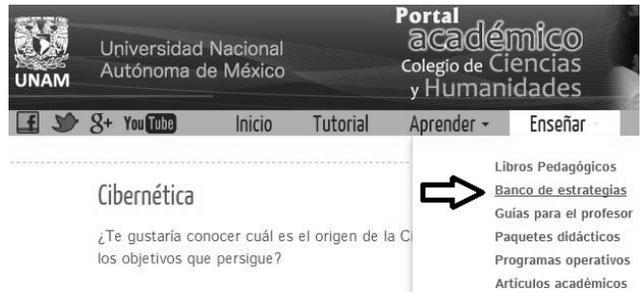
¹⁹ Diana Chávez González, *Conocimientos Fundamentales para la Enseñanza Media Superior*, <http://www.conocimientosfundamentales.unam.mx/> (consultado el 10 de septiembre del 2014)

²⁰ M. en I. Juventino Ávila Ramos, *Colegio de Ciencias y Humanidades*, <http://www.cch.unam.mx/> (consultado el 10 de septiembre del 2014)

²¹ M. en I. Juventino Ávila Ramos, *Portal Académico del Colegio de Ciencias y Humanidades*, <http://portalacademico.cch.unam.mx/> (consultado el 10 de septiembre del 2014)

²² M. en I. Juventino Ávila Ramos, *Banco de Estrategias*, <http://portalacademico.cch.unam.mx/estrategias> (consultado el 10 de septiembre del 2014)

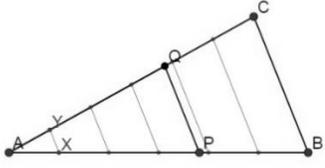
dará información acerca del tema mediante un documento *pdf*.. Como se muestra en las siguientes imágenes.



2.2.3 Proyecto Arquímedes

Es un proyecto conjunto entre el Ministerio de Educación de España y el Instituto de Matemáticas de la Universidad Nacional Autónoma de México, teniendo como propósito el desarrollo y difusión de herramientas de uso libre para la creación de unidades didácticas interactivas de Matemáticas y temas afines.

En realidad el único proyecto que sirve en línea para estudiar los antecedentes de matemáticas es Descartes JS, editor para crear escenas interactivas de matemáticas que pueden visualizarse usando Java o JavaScript para HTML 5, es un proyecto gestionado por el profesorado, donde se ofrecen recursos digitales abiertos. En la página principal del *proyecto Arquímedes*²³ (Proyecto Arquímedes, 2014) se da clic en la opción *RED Educativa Descartes*²⁴ donde encontraremos información sobre el proyecto, así como los llamados sub proyectos²⁵, utilizando la aplicación Descartes JS.

La ley de las proporciones. Eudoxio, en Elementos de Euclides, Libro V. Geometría	
$\begin{aligned} 3 \quad AX \leq AP < 4 \quad AX \quad 3 \quad AY \leq AQ < 4 \quad AY \\ 6 \quad AX \leq AB < 6 \quad AX \quad 6 \quad AY \leq AC < 6 \quad AY \end{aligned}$	<p>Teorema Si ABC y APQ son triángulos semejantes, entonces</p> $\frac{AP}{AB} = \frac{AQ}{AC}$ <p>Demostración Sea n un entero positivo y X un punto sobre AB tales que $AB = n \cdot AX$. Sea Y el punto sobre AC tal que AXY es semejante a ABC. Sea m otro entero positivo tal que</p> $m \cdot AX \leq AP < (m+1) \cdot AX$ <p>Entonces</p> $\frac{m}{n} = \frac{m \cdot AX}{n \cdot AX} \leq \frac{AP}{AB} < \frac{(m+1) \cdot AX}{n \cdot AX} = \frac{m+1}{n}$ <p>Análogamente,</p> $\frac{m}{n} = \frac{m \cdot AY}{n \cdot AY} \leq \frac{AQ}{AC} < \frac{(m+1) \cdot AY}{n \cdot AY} = \frac{m+1}{n}$
	$\frac{3}{6} \leq \frac{AP}{AB} < \frac{4}{6} \quad \left \frac{AP}{AB} - \frac{AQ}{AC} \right < \frac{1}{6}$

Desafortunadamente la información que puede ser más útil para los estudiantes, resulta tener complicaciones en la accesibilidad. Haciendo tediosa la navegación por el sitio. La siguiente dirección facilita el acceso a la página *Proyecto Descartes Miscelánea*²⁶ (Proyecto Descartes, 2014) pero para encontrarla se tiene que navegar por varios links.

²³ José Luis Abreu León, *Proyecto Arquímedes*, <http://arquimedes.matem.unam.mx/> (consultado el 10 de septiembre del 2014)

²⁴ themexpert, *RED Educativa Descartes*, <http://proyectodescartes.org/> (consultado el 10 de septiembre del 2014)

²⁵ themexpert, subproyectos, <http://proyectodescartes.org/indexweb.html> (consultado el 10 de septiembre de 2014)

²⁶ Josep M^a Navarro Canut, *Proyecto Descartes Miscelánea*, http://proyectodescartes.org/miscelanea/tematico_algebra.htm (consultado el 10 de septiembre del 2014)

2.3 La educación en línea en México.

2.3.1 Universidad Abierta y a Distancia de México²⁷ (UnADM, 2014)

Entre sitios significativos de educación en línea, se puede mencionar el de la Secretaría de Educación Pública²⁸ (SEP), en el que se puede hacer una carrera universitaria en línea de forma abierta, ya que su objetivo es formar profesionales de más alto nivel en diversas áreas del conocimiento y competitivos tanto nacional como internacionalmente. Mediante este esfuerzo la SEP busca llegar a ser la institución líder de educación superior a distancia en los países de habla hispana, con un modelo educativo flexible e innovador con alto reconocimiento social.

Se pretende ampliar las oportunidades y atender las demandas de educación superior en todo el país, bajo los estándares de calidad, con especial atención a regiones y grupos que por diversas situaciones no tienen acceso a servicios educativos escolarizados.



Se cuenta con un curso propedéutico desde el primer semestre, así como exámenes, tutores, centro virtual de lenguas y servicios escolares como se muestra en la siguiente imagen:



²⁷ SEP, Universidad Abierta y a Distancia de México, <http://www.unadmexico.mx/> (consultado el 11 de septiembre de 2014)

²⁸ Secretaría de Educación Pública, <http://www.sep.gob.mx/> (consultado el 11 de septiembre de 2014)

2.3.2 Universidad Nacional Autónoma de México y Tecnológico de Monterrey

La Universidad Nacional Autónoma de México²⁹ se ha asociado con *Coursera*³⁰ (coursera, 2014), una plataforma educativa de gran importancia en la que participan las mejores universidades del mundo. La UNAM ha aportado cuatro cursos, donde los interesados en los mismos es factible registrarse, entrar y tomarlos libremente, si se requiere la constancia del curso, ésta si se cobra, pero los cursos son libres y se muestran en las siguientes imágenes.



En los cursos proporcionados por la UNAM a *Coursera*, el único que se relaciona con el tema de tesis es el curso de *Tecnologías de información y comunicación en la educación*³¹. No se cuenta con ningún curso donde se puede desarrollar el manejo de las matemáticas.

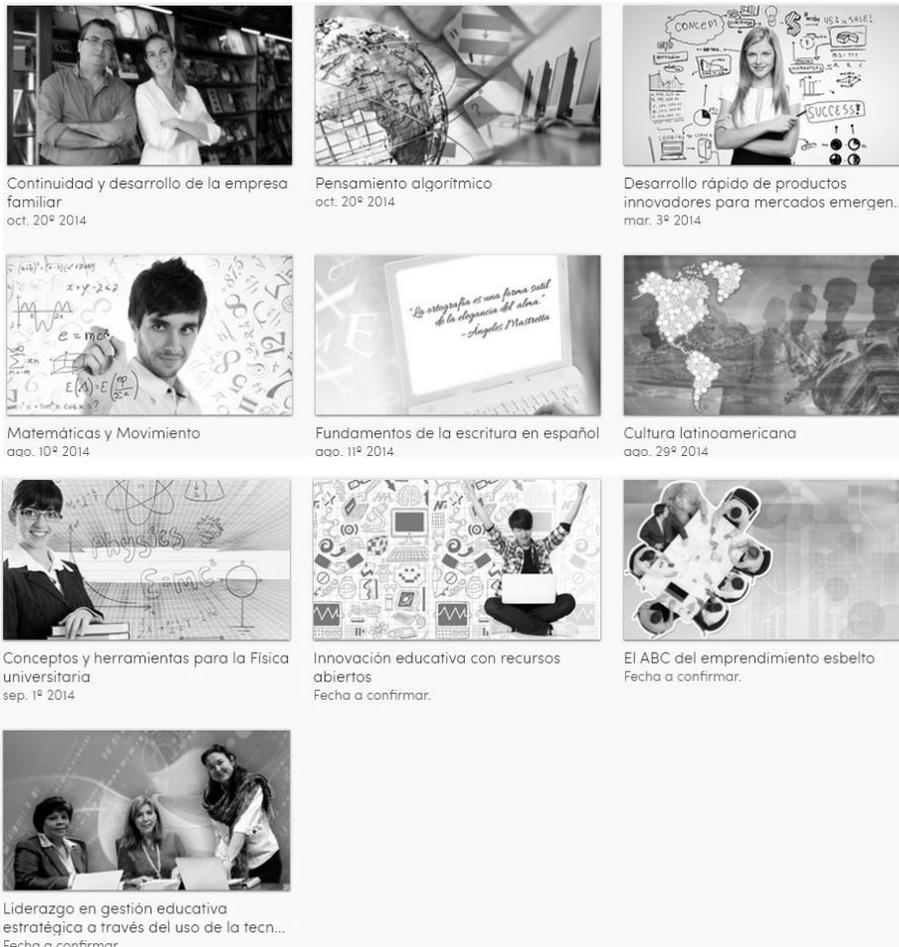
El ITESM, Instituto Tecnológico de Monterrey³² (ITESM, 2014), a pesar de ser una institución privada, comparte conocimiento que se aporta a la plataforma educativa de *Coursera* y cuenta con diez cursos que se muestran en las siguientes imágenes.

²⁹ UNAM, <http://www.unam.mx/> (consultado el 11 de septiembre de 2014)

³⁰ coursera, UNAM, <https://www.coursera.org/unam> (consultado el 11 de septiembre de 2014)

³¹ coursera, UNAM, *Tecnologías de información y comunicación en la educación*, <https://www.coursera.org/course/ticyeducacion> (consultado el 11 de septiembre de 2014)

³² Tecnológico de Monterrey, http://www.itesm.mx/wps/portal?WCM_GLOBAL_CONTEXT= (consultado el 11 de septiembre de 2014)



Cursos proporcionados por el ITESM en Coursera

El único curso que se proporciona por el ITESM sobre matemáticas es *Matemáticas y Movimiento*, se requieren cursos enfocados a la enseñanza de las matemáticas por las instituciones educativas mencionadas, ya que son muy escasos.

2.3.3 Centro de Recursos para la Enseñanza y el Aprendizaje

En el sitio de la Universidad de Guadalajara³³ (UDG, 2014) se encuentra una página Web de ayuda³⁴ (CREA, 2014) con diversos materiales educativos para fortalecer el proceso de enseñanza-

³³ Mtra. Beatriz Véliz Plascencia Universidad de Guadalajara, <http://www.udg.mx/> (consultado el 11 de septiembre de 2014)

³⁴ Dspace, Juan Raul Duran, *Centro de recursos para la enseñanza y el aprendizaje*, <http://www.crea.udg.mx/crea.jsp> (consultado el 11 de septiembre de 2014)

aprendizaje y a la formación integral de los estudiantes de nivel superior y medio superior. Los materiales de CREA ofrecen la realización de simulación de situaciones reales para solucionar problemas, enriquecen la experiencia de formación del estudiante y facilitan el proceso de enseñanza para los profesores. El espacio virtual aprovecha las TIC aplicadas a la educación desde Internet para toda la Comunidad Universitaria de Jalisco, México y el mundo.



Archivo sobre ecuaciones cuadráticas

Si se analiza los temas de Matemáticas que es de interés, nos lleva a un listado de documentos que son antecedentes del área, cómo se muestra en la siguiente imagen:

Ciencias naturales y matemáticas

En: Ciencias naturales y matemáticas

Buscar for

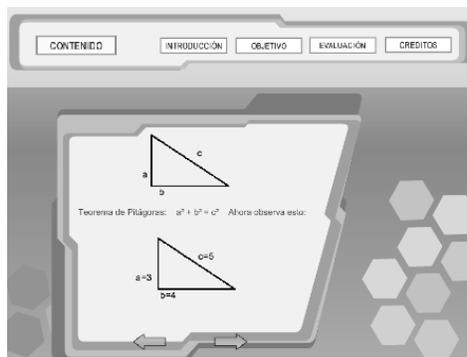
o Lista por: [Títulos](#) [Autores](#) [Materias](#) [Por fecha](#)

Colecciones en esta comunidad

- Animales (0 objetos)
- Astronomía y ciencias afines (1 objetos)
- Biología (3 objetos)
- Ciencias de la tierra (1 objetos)
- Física (16 objetos)
- Matemáticas (13 objetos)
- Paleontología, paleozoología (0 objetos)
- Plantas (0 objetos)
- Química y ciencias afines (6 objetos)

CREA es didáctica ya que permite evaluarse con la ayuda de archivos creados con un software

llamado *CREADOR*³⁵ (CREADOR, 2014), se puede observar estos materiales educativos con la ayuda de Adobe Flash Player, ya que es una aplicación en forma de reproductor multimedia creado inicialmente por Macromedia y actualmente distribuido por *Adobe Systems*. Permite reproducir archivos SWF que se crean con la herramienta de autoría como *CREADOR*.



El ejercicio de la imagen anterior se crea con la ayuda de un software de autoría de objetos de aprendizaje en línea que se llama *CREADOR*, es un interfaz gráfico que, de forma amigable, guía al usuario a través de 10 sencillos pasos para crear un Objeto de Aprendizaje. Está diseñado para facilitar la construcción de materiales educativos a personas que no son expertas en el manejo de herramientas de autoría multimedia; sin embargo, sus características le permiten a cualquier persona utilizarlo y aprovechar sus funcionalidades.



La versión 1.0 incorpora la posibilidad de incluir un tema con diferentes subtemas e imágenes de apoyo para los contenidos; además un editor para construir cuestionarios con preguntas de opción múltiple.

2.4 La educación en línea a nivel internacional

³⁵ Dspace, Juan Raúl Duran, CREADOR, <http://www.tecnologias.ciep.cga.udg.mx/creator/index.php> (consultado el 11 de septiembre de 2014)

A este nivel existen recursos que incorporan las tecnologías de la información a la educación en recursos que abarcan desde breves cápsulas educativas hasta especializaciones completas, empleando para ello una gran variedad de recursos, cada uno de ellos con limitaciones y alcances específicos. Planteando a su vez nuevos desafíos a muchas instituciones educativas, además de ofrecerles nuevas oportunidades y posibilidades. Entre ellos se hace referencia a los siguientes:

- *Wikipedia*
- *YouTube*
- *Coursera*
- *KhanAcademy*
- *Bootstrap*
- *WeScheme*

2.4.1 Wikipedia

Historia de Wikipedia³⁶ (Wikipedia, 2014). En el caso de esta enciclopedia que es totalmente libre, políglota y editada colaborativamente. Es administrada por una fundación llamada *Wikipedia*, esta sin ánimo de lucro. Tiene alrededor de 37 millones de artículos en 286 idiomas (Esta cantidad incluye dialectos de muchos de esos idiomas) y ha sido redactada en conjunto por voluntarios de todo el mundo y, por tanto cualquier persona podría editarlos.

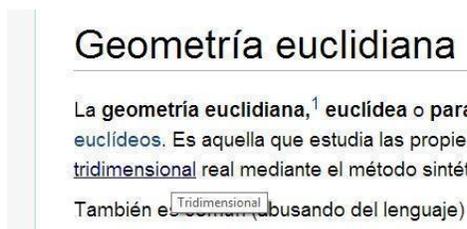


³⁶ *Wikipedia*, <http://es.wikipedia.org/wiki/Wikipedia:Portada> (consultado el 12 de septiembre de 2014)

Las Matemáticas en Wikipedia. Esta herramienta se enfoca en temas básicos de matemáticas como: Geometría y Álgebra. Como se observa en la imagen se da una pequeña explicación de lo que es la Geometría Euclidiana y se da un pequeño índice donde se puede encontrar información más específica como Axiomas, Postulados y Geometría Plana.



Algunas palabras son links que llevan a temas que son antecedentes importantes que se deben entender muy bien para poder continuar leyendo la publicación. Ejemplo de ello es la palabra “Tridimensional” que se encuentra en la lectura, donde al pasar el cursor por la palabra de color azul, de inmediato se subraya, el cursor indicará con otra imagen que se puede dar clic, ya que se trata de un link.



Se direcciona a una página relacionada con el significado de la palabra misma. En caso de existir alguna duda con respecto al significado de la palabra “Tridimensional” como se observa en la siguiente imagen.

The screenshot shows the Wikipedia article for 'Tridimensional'. At the top, there's a navigation bar with 'Artículo' and 'Discusión' tabs, and a search box. The article title 'Tridimensional' is prominently displayed. Below the title, there's a warning box about missing references. The main text explains that in physics, geometry, and mathematics, an object or entity is tridimensional if it has three dimensions, such as width, length, and depth. It also mentions that our space is tridimensional, but in reality, there are more dimensions, and it can be considered a four-dimensional space if time is included. A diagram on the right shows a 3D Cartesian coordinate system with x, y, and z axes, labeled as 'Esquema elemental de posicionamiento espacial'. On the left, there's a sidebar with various Wikipedia navigation options like 'Portal de la comunidad', 'Cambios recientes', and 'Ayuda'.

Categorías de Wikipedia. Maneja información dividida por categorías y clasificadas en forma alfabética.

Categoría:Geometría euclidiana

El artículo principal de esta categoría es: Geometría euclidiana.

- Wikimedia Commons alberga contenido multimedia sobre Geometría euclidiana.

Subcategorías

Esta categoría incluye las siguientes 8 subcategorías:

A	G (cont.)	G (cont.)
► Axiomatización de la geometría euclidiana (5 págs.)	► Geometría elemental (2 cat. 41 págs.)	► Geometría tetradimensional (13 págs.)
G	► Geometría euclidiana del espacio (1 cat. 7 págs.)	S
► Geometría convexa (15 págs.)	► Geometría euclidiana plana (3 cat. 22 págs.)	► Simetría euclidiana (13 págs.)
► Geometría del espacio (2 cat. 1 pág.)		

Páginas en la categoría «Geometría euclidiana»

Esta categoría incluye las siguientes 14 páginas:

• Geometría euclidiana	D	J
*	• Teorema de De Gua	• Teorema de Jung
• Elementos de Euclides	E	M
C	• Espacio con un número transfinito de dimensiones	• Método exhaustivo
	• Espacio euclídeo	S

En el caso de Álgebra tenemos también una gran cantidad de información y es clasificada de la siguiente forma:

- La historia del Álgebra
- Notación Algebraica
- Lenguaje Algebraico
- Estructura Algebraica

En el caso de Trigonometría.

Viene una explicación de lo que es la Trigonometría y su índice donde se aborda los siguientes temas.

- Historia

- Unidades Angulares
- Equivalencia entre las Funciones Trigonométricas
- Valor de las Funciones Trigonométricas
- Sentido de las Funciones Trigonométricas
- Representación Gráfica
- Identidades Trigonométricas

No se puede confiar en ningún escrito cuando no se sabe quién lo escribió, ya que pocos editores y contribuyentes de Wikipedia utilizan sus verdaderos nombres, y casi nunca ofrecen información sobre quiénes son. Para evaluar correctamente la información en la red, existen preguntas que se deben hacer: “¿Quién escribió esto?” y “¿Por qué lo escribieron?” En sitios con autores anónimos como Wikipedia, no es posible encontrar esta información de forma creíble. Además Wikipedia dice, “No esperamos que confíen en nosotros.”³⁷ (Wikipedia, 2014), y no se debería “utilizar Wikipedia para tomar decisiones críticas”, limitando así sus alcances.

2.4.2 YouTube

*YouTube*³⁸ (YouTube, 2014) usa un reproductor en línea basado en *Adobe Flash* para servir su contenido, aunque también puede ser un reproductor basado en el estándar *HTML5*³⁹ (2014), que *YouTube* incorporó poco después de que la *W3C*⁴⁰ (W3C, 2014) lo presentara y que es soportado por los navegadores web más importantes. Es muy popular gracias a la posibilidad de alojar vídeos personales de manera sencilla. Aloja una variedad de clips de películas, programas de televisión y vídeos musicales. A pesar de las reglas de *YouTube* en contra de subir vídeos con derechos de autor, este material existe en abundancia, así como contenidos amateur como video blogs. Los enlaces a vídeos de *YouTube* pueden ser también insertados en blogs y sitios electrónicos personales usando *API* o incrustando cierto código *HTML*.

³⁷ Wikipedia, Diez cosas que nos sabes Wikipedia, http://en.wikipedia.org/wiki/Wikipedia:Ten_things_you_may_not_know_about_Wikipedia (consultado el 12 de septiembre de 2014)

³⁸ Google, YouTube, <https://www.youtube.com/> (consultado el 12 de septiembre de 2014)

³⁹ *HTML5*, http://www.w3schools.com/html/html5_intro.asp (consultado el 12 de septiembre de 2014)

⁴⁰ W3C, <http://www.w3.org/> (consultado el 12 de septiembre de 2014)

YouTube enfocado a la educación. Es interesante experimentar que es factible aprender mejor con clases en línea por medio de videos, que en clases presenciales, ya que los primeros se pueden repetir cuantas veces sea necesario, y acompañarlos con comentarios. Muchas personas están acostumbradas a las clases presenciales, pero puede ser conveniente experimentar clases en línea, las que también pueden ser eficaces y eficientes, se evita contaminar al transportarse al colegio, se ahorra dinero para ir a la escuela y se atiende la necesidad a más estudiantes que demandan educación y que quieren culminar sus estudios, entre otros aspectos. Como se muestra en los siguientes enlaces referidos a la enseñanza de temas de Matemáticas.

Elementos de Euclides: definiciones, postulados y axiomas⁴¹

Suma de dos Ángulos interiores igual a uno externo⁴² – HD

Funciones Trigonómicas⁴³

Funciones Trigonómicas en un triángulo rectángulo⁴⁴

Sin duda *YouTube* es una gran herramienta para satisfacer las necesidades de aprendizaje de muchos estudiantes, ya que pueden encontrar una gran diversidad de cursos o videos enfocados a la educación. Pero se puede dudar si la información dada realmente es verdadera, ya que en la experiencia que se ha tenido en *YouTube* con los cursos que se han tomado, los instructores llegan a tener errores como: no tener los suficientes conocimientos, además de la carencia en el uso herramientas o gráficos, e inclusive que tan adecuada es el tipo de cámara que se maneja, etc.

Si el usuario encuentra un error en algunos de los videos del curso fue por tener conocimientos previos enfocados al tema, por lo que es recomendable leer sobre el tema con base en la mejor bibliografía y después observar los videos para reforzar los conocimientos, y siempre comentar si existe un error en el video, para que el autor realice las modificaciones pertinentes. No todos los cursos son malos o mediocres, algunos son de gran calidad, ya que el uso de herramientas como

⁴¹ YouTube, Elementos de Euclides, <http://youtu.be/bJVkaGqiKoE> (consultado el 13 de septiembre de 2014)

⁴² YouTube, Suma de dos Ángulos interiores <http://www.youtube.com/watch?v=oDA9ZN6LqPs&feature=share&list=PLC02864602575DB06> (consultado el 13 de septiembre de 2014)

⁴³ YouTube, Funciones Trigonómicas, <http://youtu.be/uMPx37LRI2E> (consultado el 13 de septiembre de 2014)

⁴⁴ YouTube, Funciones Trigonómicas en un triángulo rectángulo, <http://youtu.be/ebc9KdxBQs0> (consultado el 13 de septiembre de 2014)

pizarrones virtuales y animaciones brindan cuando se desea conocimiento de forma clara.

2.4.3 KhanAcademy

*KhanAcademy*⁴⁵ (Plataforma educativa, 2014) es una organización sin fines de lucro con el objetivo de mejorar la educación. Proporciona educación gratuita de primer nivel para cualquier persona en cualquier lugar del mundo⁴⁶.

Los recursos que maneja están disponibles para cualquiera. No importa si se es estudiante, profesor, una persona que estudia por cuenta propia, o si es un adulto que regresa a estudiar después de 20 años, todos pueden interactuar con la plataforma. Los recursos y materiales de *Khan Academy* están disponibles y completamente gratis.

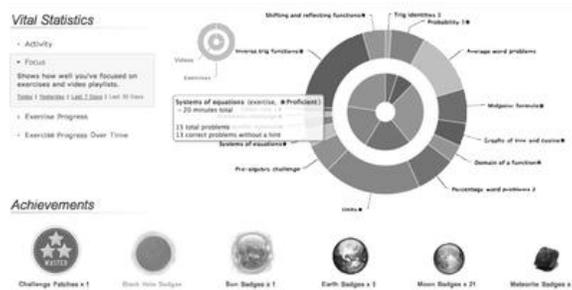


*Khan Academy*⁴⁷ cuenta con una biblioteca extensa de contenido, la cual incluye retos interactivos, evaluaciones y videos con los que se puede interactuar desde cualquier computadora con acceso a la red.

Estadísticas al instante. Se puede recordar lo que se ha aprendido y donde se está pasando el tiempo. Estos datos se mantienen en privado, ya que se brindan estadísticas para cada usuario y sus tutores. Además se puede ver sí se ha alcanzado la meta de aprendizaje y/o enseñanza propuesta.

⁴⁵ *Khan Academy*, <https://es.khanacademy.org/> (consultado el 13 de septiembre de 2014)

⁴⁶ Información sobre KhanAcademy, <https://es.khanacademy.org/about> (consultado el 13 de septiembre de 2014)



Se puede tener una vista panorámica del perfil propio al realizar cada ejercicio y problema en el que se haya trabajado. Siempre, se tiene acceso a los datos que muestran el avance y dominio de las Matemáticas por cada usuario.

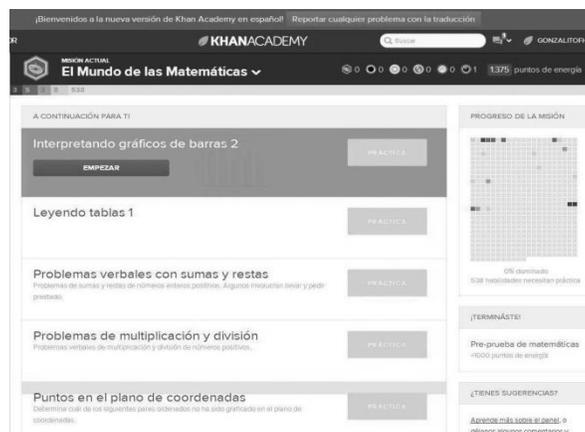
Una característica importante es que se compete con la plataforma para conseguir un triunfo personal. Se comienza con iniciar sesión, participar, y si se contesta de forma correcta se ganan puntos y medallas. Lo mejor es que se ponen retos que al resolverlos el participante se siente bien y aprende más y mejor.

Posibles medallas



Profesores y tutores pueden acceder a los datos de todos sus estudiantes. Pueden obtener un resumen del desempeño de todo el grupo o profundizar en el perfil de un estudiante para averiguar qué temas están siendo problemáticos. El reporte del tutor se permite a los tutores echar un vistazo a sus paneles de actividades y averiguar rápidamente cuál es la mejor forma de pasar su tiempo enseñando.

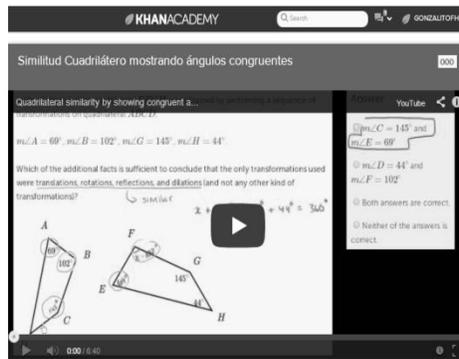
Se ofrece a los maestros el acceso a la información de sus alumnos. De inmediato se da cuenta si el estudiante tiene problemas en algunos temas en específico o si se tiene más avance que el resto de la clase.



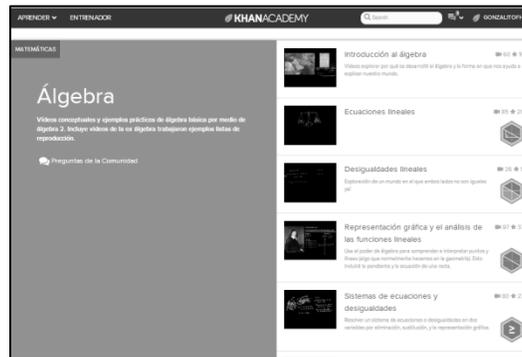
Se analiza el curso de Geometría Euclidiana en *Khan Academy*, donde se observan los temas y son bastante completos, ya que se tiene teoría y ejercicios, donde se abarca los siguientes subtemas.

- Introducción a la Geometría Euclidiana
- Ángulos y líneas de intersección
- Congruencia
- Semejanza
- Triángulos Rectángulos y Trigonometría
- Perímetro, Área y Volumen
- Círculos
- Propiedades especiales y partes de los Triángulos
- Cuadriláteros

Y dentro de estos temas se dividen en más subtemas que son explicados con videos y ejercicios.



En el caso de Álgebra también se cuenta con una lista de temas como se puede ver en la siguiente imagen.



En el caso de *KhanAcademy* se recomienda para personas que desean aprender realmente desde cero, ya que se introduce a temas desde matemáticas elementales. Sin embargo, se debe tener claro que el apoyo en otros recursos, como un buen libro de matemáticas, es fundamental para ir resolviendo dudas acerca de los ejercicios que se realizan en *KhanAcademy*. Aprender con la herramienta se muestra muy interesante, y parecería que se juega mientras se obtiene conocimientos, los recursos que se han visto durante el capítulo 2 no brindan conocimiento con la misma efectividad que *KhanAcademy* lo hace.

2.4.4 Coursera

*Coursera*⁴⁸ (2014) es una compañía educativa que decidió asociarse con las principales

⁴⁸ Página principal de Coursera, <https://www.coursera.org/> (consultado el 13 de septiembre de 2014)

universidades y organizaciones del mundo, incluyendo la UNAM⁴⁹, para poder ofrecer cursos en línea. La tecnología usada a su vez permite la posibilidad a sus asociados de llegar a millones de estudiantes.

No se parece mucho a *Khan Academy*, ya que los cursos en *Coursera* muchas veces requieren tener antecedentes previos para poder entender la información dada en los cursos. Es una plataforma educativa importante y de calidad, que imparte cursos completos, siguiendo pautas educativas de las instituciones y los profesores que los imparten, es libre y muchas veces no tiene costo, estando disponible para cualquier persona que quiera superarse.

Los cursos que se ofrecen en *Coursera* están diseñados para ayudar a dominar los materiales, los profesores son altamente capacitados y sobre todos se puede aprender al propio ritmo y comprobar los conocimientos que se aprenden, reforzándolos a través de ejercicios interactivos.

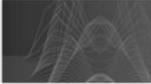
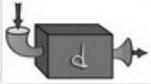
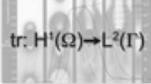


En lo que se refiere a temas de Matemáticas, existe un buscador en *Coursera* por categorías donde una de ellas es Matemáticas y donde se localizan 59 cursos⁵⁰ en diferentes idiomas.

⁴⁹ Unam asociado a Coursera, <https://www.coursera.org/unam> (consultado el 13 de septiembre de 2014)

⁵⁰ coursera, cursos sobre matemáticas, <https://www.coursera.org/courses?orderby=upcoming&cats=math> (consultado el 11 de septiembre de 2014)

Cursos

	Universidad Estatal de Ohio Cálculo Uno con Jim Fowler	Avanza a tu propio ritmo. ¡Aprende ya mismo!
	Universidad de Princeton Análisis de algoritmos con Robert Sedgewick	sep. 5ª 2014 6 weeks de duración
	Universidad de Pensilvania Cálculo: en una variable con Robert Ghrist	sep. 8ª 2014 13 weeks de duración Certificado Verificado
	Escuela Central Paris Una introducción al análisis funcional con John Cagnol & Anna Rozanova-Pierrat	sep. 12ª 2014 8 weeks de duración Certificado Verificado
	Universidad Autónoma de Barcelona Precálculo con Jaume Pujol & Mercè Villanueva	sep. 15ª 2014 9 weeks de duración Certificado Verificado

Los cursos provienen de diferentes universidades del mundo como se observa en la imagen anterior, se muestran características como: las semanas que duran, en qué fecha van a iniciar o si en el instante se puede comenzar, los profesores que lo van a impartir y el título del mismo curso.

Se puede ver el curso totalmente gratis pero cuando se requiera de reconocimiento certificado entonces se puede obtener a un costo un certificado.



The screenshot shows the Coursera interface for a verified certificate. At the top, it says "Obtén un Certificado Verificado". Below this, there is a section for "UCIRVINE" with the course title "Pre-Calculus: Series, Extrema and Riemann-Stieltjes Integrals" and the instructor "Dustin Zorn". A note states: "El Programa de Reconocimiento Verificado es necesario para la inscripción". At the bottom, there are three icons: a person icon for "Check el Programa de Reconocimiento", a certificate icon for "Obtén tu Certificado", and a LinkedIn icon for "Comparte tu éxito".

En *Coursera* se estudian cursos completos acerca de temas hasta hora más avanzados donde se da por hecho que Álgebra y Geometría se tienen dominados. Este MOOC es una excelente herramienta, con una forma de enseñanza-aprendizaje diferente a los recursos mencionados

previamente. Los cursos de plataformas como *Coursera* son más confiables que los que aparecen en *YouTube*, ya que provienen de universidades de renombre un ejemplo de ellos: Universidad Nacional Autónoma de México, Universidad de Michigan⁵¹ (Michigan, 2014), Universidad de Tokio⁵² (Tokio Coursera, 2014) entre otras asociadas con *Coursera*.

2.4.5 *Bootstrap*

Es un excelente sitio⁵³ (*Bootstrap*, 2014) para aprender una buena parte del currículo de matemáticas de nivel medio (últimos años de primaria, secundaria y preparatoria), con énfasis en conceptos algebraicos, geométricos y trigonométricos mediante programación, la relevancia de ese sitio se puede verificar de muchas maneras, una de ellas realizando una búsqueda en Google con los términos *Algebraic Video Game Programming* las primeras ocho opciones que surgen de tal búsqueda hacen referencia al mismo⁵⁴ (Búsqueda en línea, 2014).

Bootstrap funciona principalmente mediante módulos, los que al terminar cada uno, el estudiante obtiene un libro de trabajo completo con problemas verbales, notas y desafíos matemáticos, así como un video juego diseñado por el mismo. La motivación y la confianza del estudiante despertados por este diseño se aplica directamente al aprendizaje de matemáticas a fin de crear algo muy atractivo para el estudiante.

Esta web no sólo es un recurso valioso en Internet, sino que es un medio integral de apoyo a la enseñanza-aprendizaje de matemáticas, pues es una forma divertida para que los estudiantes aprendan y apliquen conceptos matemáticos y principios rigurosos de programación a fin de crear un videojuego sencillo. Todos los materiales que se emplean están alineados con el *Common Core Standards for Mathematics*⁵⁵ (Standards for Mathematics, 2014), incluyendo los nuevos estándares

⁵¹ Universidad de Michigan, <https://www.coursera.org/umich> (consultado el 13 de septiembre de 2014)

⁵² Universidad de Tokio, <https://www.coursera.org/utokyo> (consultado el 13 de septiembre de 2014)

⁵³ *Bootstrap*, página principal, <http://www.bootstrapworld.org/> (consultado el 13 de septiembre de 2014)

⁵⁴ Búsqueda de métodos de enseñanza algebraica con programación de juegos, <https://www.google.com.mx/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=Algebraic%20Videogame%20Programming> (consultado el 27 de septiembre de 2014)

⁵⁵ Common Core Standards for Mathematics, <http://www.corestandards.org/Math/> (consultado el 13 de septiembre de 2014)

de la *Mathematical Practice*⁵⁶ (Standards for Mathematical Practice, 2014).

Esta web, a la vez que metodología, así como medios de apoyo al profesorado están a cargo de Emmanuel Schanzer con base en el trabajo de Matthias Felleisen⁵⁷ y el proyecto *Program by design*⁵⁸. Se cuenta con talleres de ayuda para profesores que enseñan matemáticas y programación, es una ingeniosa y atractiva forma de enseñanza, ya que se aprende creando un videojuego. El sitio web se muestra en la siguiente imagen.



Bootstrap cuenta con ayuda tanto a estudiantes y a profesores con ganas de aprender y enseñar.

Entre sus principales características están el ser:

Orientado a estudiantes: se profundiza sobre la importancia y relación que tienen las matemáticas, como Aritmética, Álgebra, Geometría y Trigonometría, entre otras, con la programación de computadoras a través de un currículum correlacionado con los contenidos de matemáticas de enseñanza media, evaluando la práctica con ejercicios propuestos por los integrantes de *Bootstrap*

Orientado a profesores: con *Bootstrap* existe una forma particular de enseñanza de la programación mediante emplear conceptos como los algebraicos como base para crear imágenes, animaciones y videojuegos, comenzando desde cero. Implementado en referencia a los *Standards*

⁵⁶ Standards for Mathematical Practice, <http://www.corestandards.org/Math/Practice/> (consultado el 13 de septiembre de 2014)

⁵⁷ Matthias Felleisen, Información sobre Matthias Felleisen, <http://www.ccs.neu.edu/home/matthias/> (consultado el 13 de septiembre de 2014)

⁵⁸ Program by Design, <http://www.programbydesign.org/> (consultado el 13 de septiembre de 2014)

*for Mathematical Practice*⁵⁹ (Estandares, 2014), por lo que *Bootstrap* se construye desde un enfoque pedagógico que brinda una solución a problemas que inicialmente se dan de forma verbal y difusa, y de forma iterativa permiten llegar a una solución precisa mediante lo que denomina Recetas de Diseño. De esta forma los estudiantes resuelven problemas que se dan de forma escrita mediante crear un videojuego.

Los profesores cuentan con dos módulos de enseñanza, el primer módulo llamado *Bootstrap:1*⁶⁰ el cual se enfoca a que los estudiantes aprendan y apliquen conceptos matemáticos y principios de programación para crear un sencillo videojuego. En el caso del módulo llamado *Bootstrap:2*⁶¹, enfocándose más de lleno en la aplicación de las matemáticas a medios más avanzados de programación, teniéndose así la libertad de que los estudiantes puedan crear juegos más avanzados donde utilicen estructuras de datos y eventos más elaborados.

Donde trabajar con *Bootstrap*. Si bien todo el material de *Bootstrap* está en línea de forma abierta suficientemente documentado, los profesores y alumnos cuentan con varias sedes en los Estados Unidos donde pueden acudir personalmente y online a talleres. Se deja indicado que si en el lugar donde se radica no existe *Bootstrap*, inmediatamente se informa que se visite el área de la ciudad propuesta dentro de los Estados Unidos, para que se trabaje en esto.

Si se completan los módulos de *Bootstrap* 1 y 2 puede continuarse trabajando con material más complejo como *Picturing Programs*⁶², *How to Design Programs*⁶³ y *Realm of Racket*⁶⁴ (Realm of Racket, 2014).

⁵⁹ Estándares que maneja Bootstrap, <http://www.bootstrapworld.org/materials/Spring2014/Standards.shtml> (consultado el 13 de septiembre de 2014)

⁶⁰ Bootstrap, Modulo Bootstrap: 1, <http://www.bootstrapworld.org/materials/Fall2014/courses/bs1/main.shtml> (consultado el 13 de septiembre de 2014)

⁶¹ Bootstrap, Modulo Bootstrap: 2, <http://www.bootstrapworld.org/materials/Fall2014/courses/bs2/main.shtml> (consultado el 13 de septiembre de 2014)

⁶² Picturing Programs, <http://picturingprograms.com/> (consultado el 13 de septiembre de 2014)

⁶³ How to Design Programs, Second Edition, <http://www.ccs.neu.edu/home/matthias/HtDP2e/> (consultado el 13 de septiembre de 2014)

⁶⁴ Realm of Racket, <http://www.realmofracket.com/> (consultado el 13 de septiembre de 2014)

CAPÍTULO 3

Bootstrap y el aprendizaje de Matemáticas

3.1 Talleres para la enseñanza

Entre los apoyos de *Bootstrap*⁶⁵ (Bootstrap, 2014) a la enseñanza de matemáticas mediante videojuegos Los profesores pueden solicitar talleres para impartirse en su escuela, llenando un formulario de solicitud.

En un taller de *Bootstrap* se hace lo siguiente:

- Revisar el currículo
- Se explora la investigación en la que se apoya currículo
- Investigar los problemas cognitivos de estudiantes que tienen problemas con Matemáticas
- Ser un estudiante por un día y ver las lecciones impartidas por un profesor capacitado
- Ganar créditos académicos.

En un taller de *Bootstrap* se trabaja con profesores para discutir experiencias de clase y pasar el día en los zapatos del alumno. Se tiene oportunidad de probar las actividades de cada clase *Bootstrap*. Se aprende a informar a otros profesores, se habla de pedagogía con los instructores y se prueban los materiales y el software. Finalmente, se crea un videojuego.

Conocer las bases de la investigación del proyecto. Se exploran los fundamentos de Álgebra, y por qué de la diferencia con la Aritmética y que tan importante son los temas para el desarrollo de todo estudiante. En el taller de *Bootstrap* se tiene oportunidad de utilizar información actualizada de los estudios de Matemáticas y programación aplicándolo a asignaturas de nivel medio.

Sus módulos. Los *Módulos Bootstrap* se emplean para impartir clases de Matemáticas y Programación para profesores con limitada experiencia enseñando Informática. Las personas que pertenecen a este proyecto implementan el desarrollo profesional con talleres de capacitación a docentes, tanto de contenido como pedagógico a fin de poder entregar conocimiento de forma

⁶⁵ Bootstrap, Página principal, <http://www.bootstrapworld.org/> (consultado 22 de septiembre de 2014)

eficaz.

Los materiales se dividen en dos:

Bootstrap:1

Si se requiere aplicar conceptos matemáticos y principios de programación para crear un videojuego la opción es *Bootstrap1*. Todas las clases se alinean a las normas fundamentales comunes para las matemáticas, incluyendo los nuevos estándares para la práctica de Matemáticas (de algún estado de la Unión Americana).

Bootstrap:2

Si se quiere profundizar más en programación y se desea que los estudiantes empiecen a crear juegos más avanzados, se debe enfocar en el siguiente módulo llamado *Bootstrap:2*. Si además se busca una verdadera clase de programación y se desea que estudiantes tengan la libertad para crear juegos más avanzados utilizando estructuras de datos, eventos adicionales y mucho más, se querrá seguir el segundo módulo (*Bootstrap: 2*), construyendo sobre lo que han hecho los estudiantes en el curso introductorio. Se ha listado el desglose de los conceptos en la imagen de abajo, donde se puede encontrar el mejor ajuste para una clase.

Las siguientes imágenes son un listado de los conceptos, tanto de *Bootstrap:1* y *Bootstrap:2*

<p>A partir de Bootstrap: 1...</p> <p><i>Los estudiantes crean un juego simple, 3 caracteres que implican un jugador, un blanco y un peligro. Diseñan lo que parece a cada personaje y utilizan conceptos algebraicos para detectar colisiones, manejar las pulsaciones del teclado y determinar cómo se mueven e interactuar.</i></p>	<ul style="list-style-type: none">• Problemas verbales• Coordinar aviones• Orden de operaciones• Variables• Funciones• Tablas de la función• Dominio y rango• Función composición• Desigualdades• Funciones por trozos• Teorema de Pitágoras• Número de líneas• Polígonos	<ul style="list-style-type: none">• Números, las cadenas y las imágenes• Definición de funciones• Pruebas unitarias• Lógica booleana• Funciones multientradas• Funciones de tipo mixto
---	---	---

<p>Continuando a Bootstrap: 2...</p> <p><i>Los estudiantes aprenden cómo el bucle de eventos basados en el mundo que mueve sus manos a la obra 1 obras del juego y utilizarlo para crear animaciones utilizando tipos de datos simples para su mundo. Luego, aprender acerca de las estructuras de datos y diseñar una estructura del mundo para un sofisticado juego de su propio diseño.</i></p>	<ul style="list-style-type: none"> • Complejas relaciones funcionales • Explorando la aleatoriedad • Conexiones a trigonometría 	<ul style="list-style-type: none"> • Programación orientada a eventos • Estructuras de datos • Todo-programa de diseño • Modelado de datos • Encapsulación • Conexiones de recursividad, listas y algoritmos
---	--	--

Se presenta un libro de trabajo (*Student Workbook*) para alumnos, y se encuentra disponible por petición uno específico para profesores, ambos relacionados con las unidades que se muestran en el sitio de *Bootstrap*.

Dentro de esta Unidad 1 y las demás unidades, se encuentran notas dadas por los profesores, así como una página de discusión de grupo de noticias.

The screenshot shows a web interface for a discussion board. At the top, there are navigation links: "me back to Bootstrap! To start off, you're going to practice everything Bootstrap 1. How much do you remember?", "Introduce teaching staff. Give some background: age, where you're from.", "Hide Teacher Notes", and "Discussion Group". Below these are buttons for "Marcar todo como leído" and "Filtros". A search bar with a "Buscar" button is present. The main content area shows a list of posts:

- Student Workbooks are now on sale!** (1) De Schanzer (Director) - 1 publicación - 32 vistas 8 ago
- December Curriculum Updates** (3) De Schanzer (Director) - 3 publicaciones - 94 vistas 30 mar
- Uploading Image Files** (3) De Schanzer (Director) - 3 publicaciones - 4 vistas 21 sep
- Is there a polygon function** (5) De madge...@asdnh.org - 5 publicaciones - 10 vistas 17 sep

- Unidad 1: Los estudiantes discuten los componentes de sus videojuegos favoritos, y descubren que pueden reducirse a una serie de coordenadas. A continuación, exploran las coordenadas en el espacio cartesiano, e identifican las coordenadas de los personajes en un juego

en varios puntos en el tiempo. Una vez que se sientan cómodos con coordenadas, construyen listas de coordenadas para los diferentes momentos en su propio juego.

La agenda que tiene la unidad menciona cuánto es el tiempo en que se tarda en ver los temas mencionados en la misma, los temas vistos en la primera unidad son los siguientes:

Agenda

Introduction	5 min
Expressions	30 min
Contracts	20 min
The Design Recipe	30 min
Closing	5 min

- Unidad 2: Los estudiantes son introducidos a una representación de conjunto de mapeo de funciones, el objeto existe en función como un medio de traducir los puntos de un dominio en un rango. Junto con su comprensión de los Círculos de Evaluación, los estudiantes generalizan su comprensión de las funciones para incluir otros tipos de datos, incluyendo cadenas e imágenes.

Agenda

Review	20 min
Introducing Structs	10 min
Autos	20 min
Accessor Functions	10 min
Autobody Shop	25 min
Closing	5 min

- Unidad 3: Los estudiantes son introducidos a la ventana Definiciones y aprender la sintaxis para la definición de los valores de varios tipos. También se introducen a la sintaxis de la definición de funciones y la creación de ejemplos.

Agenda

Review: the Autobody Shop	20 min
define-struct	5 min
The Party Struct	30 min
Acting Out Ninja World	30 min
Closing	5 min

- Unidad 4: Los estudiantes continúan practicando la Receta Diseño aplicándolo a problemas sencillos.

Agenda

Review	5 min
Codewalking	20 min
Extending the World	15 min
Game Brainstorming	15 min
Game Design!	10 min
Defining the World	20 min
Closing	5 min

- Unidad 5: Los estudiantes definirán las funciones que se asignan a los atributos de su juego a partir de un fotograma al siguiente, lo que les permite mover sus peligros, las metas y los proyectiles.

Agenda

Introduction	5 min
Drawing the World	40 min
Updating the World	40 min
Closing	5 min

- Unidad 6: Los estudiantes descubren tipos booleanos, y los utilizan para crear programas que ponen a prueba los valores, modelan escenarios con el uso de estos programas.

Agenda

Introduction	15 min
Keypress	35 min
Keypresses in Your Game	35 min
Closing	5 min

- Unidad 7: Los estudiantes usan la Geometría y el condicional para mover sus personajes jugadores en respuesta a las presiones de teclas.

Agenda

Introduction	5 min
Protecting the Boundaries	20 min
Tests and Results	45 min
Branches in update-world	15 min
Closing	5 min

- Unidad 8 : Los estudiantes obtienen, discuten y demuestran el Teorema de Pitágoras

Agenda

Introduction	5 min
1D Distance	10 min
The Distance Formula	20 min
Collide?	10 min
update-world	40 min
Closing	5 min

- Unidad 9: Los estudiantes editan detalles del juego y se preparan para su fiesta de lanzamiento

Agenda

Introduction	5 min
Randomizing Ninja World	15 min
Scoring and Levels	35 min
Challenge: Nested Structures	30 min
Closing	5 min

- Lecciones complementarias: para los profesores en busca de ejercicios adicionales, donde se han recopilado muchas actividades para que los estudiantes vayan más profundamente en la materia.

Existen opciones disponibles para el aprendizaje y se encuentran en la página de *Racket*⁶⁶ (Racket, 2014) para profundizar en los temas, por lo que se cuenta con más recursos para los interesados:

*Picturing Programs*⁶⁷ (Picturing Programas, 2014)

Se centra en imágenes y animaciones, los estudiantes exploran la recursividad, listas, árboles y algoritmos utilizando un mismo lenguaje y una receta de diseño que están disponibles en la página de *Bootstrap*.

*How to Design Programs*⁶⁸ (How to Design Programs, 2014)

Un enfoque basado en los principios para el diseño de programas y que puede servir de apoyo en la enseñanza del lenguaje utilizando el ambiente de programación *DrRacket*, dirigido a todas aquellas personas que no tienen ningún conocimiento sobre programación, o desean ampliar sus conocimientos de programación.

*Realm of Racket*⁶⁹ (Realm of Racket, 2014)

Si se es estudiante de Ingeniería en Computación, o áreas afines, y se desea aprender *Racket* y diseño de programas creando juegos, cuyo código se crea con la distribución de *Racket*, para los que desean iniciarse con *Racket*.

⁶⁶ Racket, <http://racket-lang.org/> (consultado el 22 de septiembre de 2014)

⁶⁷ Picturing Programs, <http://picturingprograms.com/> (consultado el 22 de septiembre de 2014)

⁶⁸ How to Design Programs, <http://www.htdp.org/> (consultado el 22 de septiembre de 2014)

⁶⁹ Realm of Racket, <http://www.realmofracket.com/> (consultado el 22 de septiembre de 2014)

*Programming Languages Application and Interpretation*⁷⁰ (PLAI, 2014)

Fundamentos de lenguajes de programación, si se desea comprender las características que hacen funcionar a los lenguajes, enfocado a estudiantes universitarios, estudiantes de posgrado y expertos.

*Semantics Engineering with PLT Redex*⁷¹ (PLT Redex, 2014)

Automatización ligera de semánticas de lenguajes de programación y modelaje de semánticas de lenguajes de programación, para el ingeniero en lenguajes de programación.

Se cuenta también con diferentes recursos en la web que facilitan información necesaria para aprender más sobre el lenguaje. Existe una página llamada *GitHub*, que ofrece cursos⁷² (GitHub, 2014) referentes a *Racket* proporcionados por diferentes universidades y colegios de los Estados Unidos como son: *Indiana University*⁷³ (Indiana, 2014), *Northeastern University* (Northeastern, 2014), *Tufts University*⁷⁴ (Tufts, 2014), *Vassar College*⁷⁵ (Vassar, 2014), *University of British Columbia*⁷⁶ (CPSC 110, 2014), *University of New Brunswick*⁷⁷ (CS3613: Programming Languages, Winter, 2014), *Berry College*⁷⁸ (Computer Science Moodle, 2014), *Coursera* (ProgramDesign, 2014), *University of Waterloo* (CS 115, 2014) además de 6 cursos y *University of Utah* (CS 1410-20: Object-Oriented Programming, 2014).

Se realizan reuniones como la ocurrida en *St.Luis* el pasado 20 de septiembre del 2014, donde se

⁷⁰ PLAI, <http://cs.brown.edu/~sk/Publications/Books/ProgLangs/2007-04-26/> (consultado el 22 de septiembre de 2014)

⁷¹ PLT Redex, <http://redex.racket-lang.org/> (consultado el 22 de septiembre de 2014)

⁷² GitHub, Cursos de Racket, <https://github.com/plt/racket/wiki/Courses-using-Racket> (consultado el 22 de septiembre de 2014)

⁷³ Curso de Racket, Universidad de Indiana, <http://www.cs.indiana.edu/classes/c211/> (consultado el 27 de septiembre de 2014)

⁷⁴ Universidad de Tufts, COMP 50, <http://www.cs.tufts.edu/comp/50PSS-2013f/> (consultado el 27 de septiembre de 2014)

⁷⁵ Colegio Vassar, C.S. Dept. Courses <http://www.cs.vassar.edu/courses/top> (consultado el 27 de septiembre de 2014)

⁷⁶ University of British Columbia, CPSC 110, <https://sites.google.com/site/ubccpsc110/> (consultado el 27 de septiembre de 2014)

⁷⁷ CS3613: Programming Languages, Winter 2014, <http://www.cs.unb.ca/~bremner/teaching/cs3613/> (consultado el 27 de septiembre de 2014)

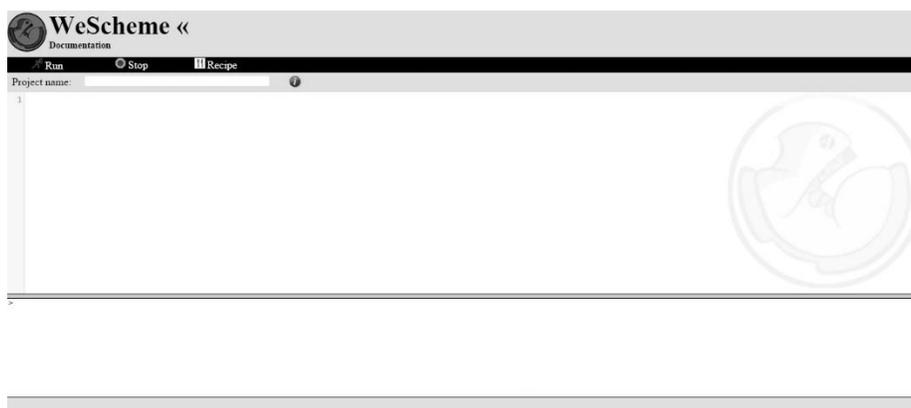
⁷⁸ Computer Science Moodle, <https://cs.berry.edu/moodle/> (consultado el 27 de septiembre de 2014) (consultado el 27 de septiembre de 2014)

llevó a cabo el *RacketCon*⁷⁹ (Reunion, 2014), y se trata de una reunión pública para todos los interesados del lenguaje *Racket* como: desarrolladores, programadores y educadores. Es la oportunidad para de que todos los miembros de la comunidad se reúnan y compartan planes e ideas para ayudar así al futuro del lenguaje *Racket*.

Empresas que apoyan el proyecto. *Bootstrap* cuenta con patrocinio de empresas, su apoyo es mediante voluntarios y financiero, entre los patrocinadores están: Apple, Cisco, la Asociación de Software de Entretenimiento (ESA), *Facebook*, *Google*, Fondo de donaciones caritativas Google Inc., Fundación Tides, IBM, Jane Street Capital, LinkedIn, Microsoft, la Fundación Nacional para la Ciencia, NVIDIA, Thomson / Reuters, y personas que realizan donaciones privadas.

¿Qué ambiente de programación se utiliza?

Bootstrap utiliza el ambiente de programación *WeScheme*, un IDE basado en la nube que no requiere de instalación, ya que se encuentra montado en la web. Para almacenar archivos en la nube se cuenta con un registro previo donde se pide una cuenta de Gmail. Funciona con cualquier navegador. Se comparte simplemente mediante enviar un vínculo o mensaje, o al publicar en sitios como Facebook, Twitter, Reddit, Código QR, etc. El ambiente de programación se muestra en la siguiente imagen.



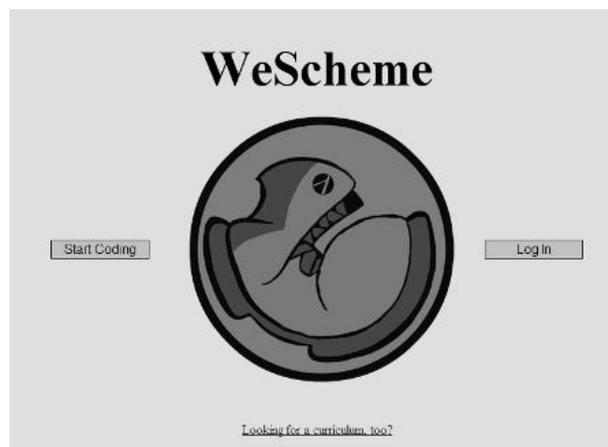
⁷⁹ RacketCon, <http://con.racket-lang.org/> (consultado el 27 de septiembre de 2014)

Los programas escritos en *WeScheme* están disponibles desde cualquier dispositivo, con un navegador habilitado con *JavaScript*. Así se evitan los problemas como sucede al instalar aplicaciones en las distribuciones de Linux, ya que *WeScheme* contiene el compilador, el entorno de edición y las bibliotecas de tiempo de ejecución.

Se puede obtener la versión instalable para correr código en una computadora de escritorio, para ello se tiene que instalar el ambiente de programación *DrRacket*, un entorno multiplataforma soportado en diversos sistemas (*Windows, OS X, Unix/Linux*).

3.2 Interactuando con el entorno de programación.

Para iniciar con el ambiente se abre un navegador y se dirige a la siguiente dirección <http://www.wescheme.org>⁸⁰ (2014)



Se presiona el botón de la izquierda “*Start Coding*”. En la siguiente página⁸¹ (Ambiente de programación, 2014) se encuentra dividida en dos secciones, la sección superior es la ventana de definiciones y la sección inferior es la ventana de interacciones como se observa en la imagen siguiente, marcados sus nombres en inglés en la esquina inferior derecha. Se observan también las

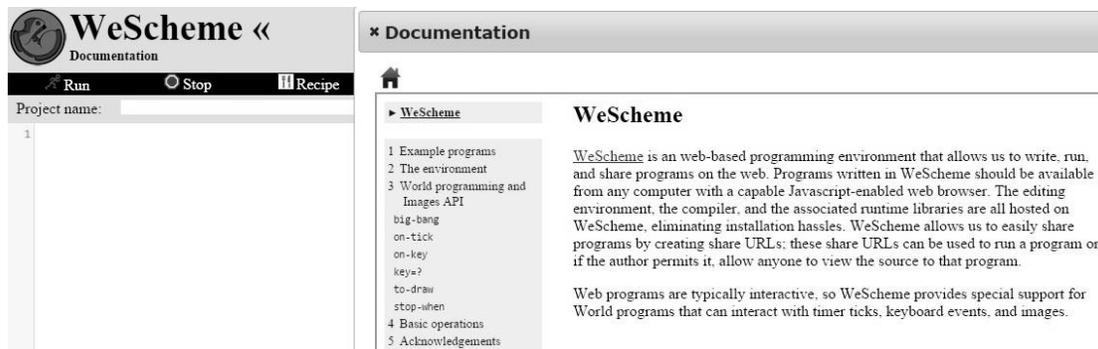
⁸⁰ Danny Yoo, *WeScheme*, <http://www.wescheme.org> (consultado el 19 de septiembre de 2014)

⁸¹ Danny Yoo, Ambiente de programación, <http://www.wescheme.org/openEditor> (consultado el 19 de septiembre de 2014)

opciones que se incorporan en la parte superior del entorno, *Run*, *Stop*, *Recipe* y *Project name*.



Se aprecia en la siguiente imagen dando clic en el link *Documentación* donde se encuentra: información, operadores básicos, programación de mundos y programas como ejemplos, que describen características importantes del ambiente y el lenguaje *Racket*⁸² utilizado.



Características del lenguaje *Racket* con el uso de *WeScheme*. Es importante saber el manejo del lenguaje para poder entender y desarrollar ejercicios. No es necesario ser experto en *Racket* (Lenguaje, 2014), ya que solo se requiere lo básico para cumplir con el aprendizaje de las Matemáticas.

⁸² Lenguaje de programación, <http://racket-lang.org/> (consultado el 19 de septiembre de 2014)

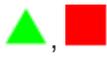
Los tipos de datos que se van a manejar con la herramienta *WeScheme* son los siguientes:

Tipo de datos

Números 1, 3.5, 01.02, # i1.414213562373095

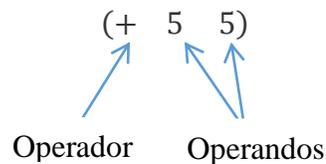
Booleanos true, false

Cadenas "Marvolo", "Negro", "zanahoria"

Imágenes 

La forma de escribir expresiones en *Racket* es la siguiente:

- Abrir paréntesis
- Escribir el operador primitivo
- Se escriben los operandos
- Se cierra el paréntesis



Es una llamada primitiva, porque comienza con un paréntesis de apertura, y un operador primitivo.

(* 4 5)

Se puede tener expresiones más elaboradas conforme se avance como lo ejemplifica la siguiente expresión.

(- (+ (* 12 8) 17) 8)

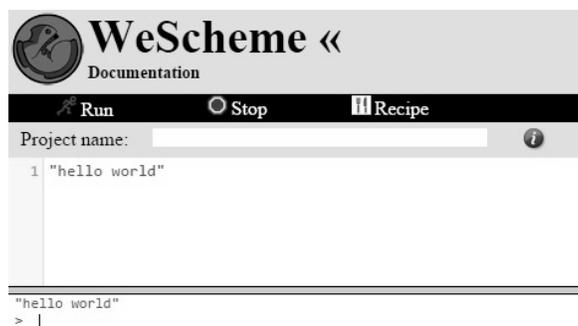
Existen otros operadores: `(abs x)`, `(sub1 z)`, `(add1 z)`, `(modulo n m)`, `(quotient n m)`, `(remainder n m)`, `(gcd n)`, `(numerator q)`, y `(denominator q)`, `(sqrt z)`, `(expt z w)`, `(exp z)` y `(log z)`, se tiene también funciones trigonométricas como: `(sin z)`, `(cos z)`, `(tan z)`, `(asin z)`, `(acos z)` y `(atan z)`, existen también los operadores de comparación con números donde se obtiene por respuesta un booleano como los siguientes: `(= z w)`, `(< x y)`, `(<= x y)`, `(> x y)` y `(>= x y)`.

Un ejemplo sencillo del resultado que produce una cadena en el ambiente *WeScheme*, es el típico ejemplo *Hello World*.

Se da clic sobre la ventana definiciones y se introduce el siguiente tipo de dato llamado cadena:

```
"hello world"
```

Al haber escrito lo siguiente, inmediatamente se da un clic en el botón *Run* en la barra de herramientas de la parte superior. Si se cumple con la sintaxis, se debe ver "hello world" en la ventana de interacciones, la siguiente imagen lo muestra.



En el caso de las imágenes se utilizan las que se encuentren en la web y se detectan como valores, como también *cadena*, *números*, *booleanos* y *estructuras*". Pudiéndose aplicar operaciones aritméticas y de álgebra en ellos. La siguiente imagen muestra la forma en que se puede operar con una imagen en el siguiente ejemplo:

En la ventana de definiciones si se agregan las siguientes líneas de código:

```
(bitmap/url "http://racket-lang.org/logo.png")  
(rotate 45 (bitmap/url "http://racket-lang.org/logo.png"))
```

Se obtienen las imágenes siguientes:



En el ejemplo siguiente se usan dos imágenes que se encuentran en la red:

<http://www.wescheme.org/css/images/BigLogo.png>⁸³



⁸³ Imagen del logo WeScheme, <http://www.wescheme.org/css/images/BigLogo.png> (consultado el 19 de septiembre de 2014)

<http://racket-lang.org/logo.png>⁸⁴



Con las imágenes se utiliza un *overlay*, donde se superpone una imagen encima de otra y a su vez se tiene por respuesta la transformación en una sola imagen, en el caso del ejemplo siguiente, el logo de *Racket* se queda encima del logo de *WeScheme*.

```
(overlay (bitmap/url "http://racket-lang.org/logo.png")  
         (bitmap/url "http://www.wescheme.org/css/images/BigLogo.png"))
```



Para el caso de imágenes se cuenta con más operadores, como los siguientes: *crop*, *frame*, *above*, *beside*, *scale/xy*, *flip horizontal*, *flip vertical*, *scale*, *rotate*, *place-image/align*, *place-image*, *above/align*, *above*, *beside/align*,

⁸⁴ Imagen del logo de Racket, <http://racket-lang.org/logo.png> (consultado el 19 de septiembre de 2014)

`underlay/xy`, `underlay/align`, `underlay`, `overlay/xy`, `overlay/align`, `overlay`, `polygon`, `regular-polygon`, `radial-star`, `star-polygon` y otros operadores para dibujar figuras geométricas conocidas como, `star`, `rhombus`, `rectangle`, `square` y en el caso de dibujar triángulos se realiza de manera diferente, ya que los triángulos cambian las características de sus argumentos por ejemplo:

Se requiere para el primer triángulo dibujar el valor de uno de sus lados y dos ángulos, pero en el segundo ejemplo se requiere conocer dos ángulos y un lado

```
(triangle/saa 200 10 40 "solid" "seagreen")
```



```
(triangle/aas 130 40 40 "solid" "lightseagreen")
```



Todas las formas de dibujar triángulos son las siguientes: `triangle/saa`, `triangle/asa`, `triangle/aas`, `triangle/ssa`, `triangle/sas` y `triangle/ass` donde se vieron ejemplos de dos previamente, se puede crear también imágenes con letras con la función `text`.

Algunas funciones como `scene+polygon`, `make-pen` solo funcionan en el ambiente de programación instalable ya que *WeScheme* no es tan completo como *DrRacket*. En el caso de *DrRacket* se utiliza `require` y se importan funciones de otros módulos, ya que si obtenemos esas

funciones facilitándose realizar con éxito ejercicios que requieran de esas funciones como se muestra en el siguiente ejemplo de imágenes.

En el caso del ambiente de programación *DrRacket*

```

;;(require 2htdp/image)

(underlay
 (rectangle 80 80 "solid" "mediumseagreen")
 (polygon
  (list (make-posn 0 0)
        (make-posn 50 0)
        (make-posn 0 50)
        (make-posn 50 50))
  "outline"
  (make-pen "darkslategray" 10 "solid" "round" "round")))

(require 2htdp/image)

(underlay
 (rectangle 80 80 "solid" "mediumseagreen")
 (polygon
  (list (make-posn 0 0)
        (make-posn 50 0)
        (make-posn 0 50)
        (make-posn 50 50))
  "outline"
  (make-pen "darkslategray" 10 "solid" "round" "round")))

```

Welcome to **DrRacket**, version 6.0.1 [3m].
 Language: Beginning Student [custom]; memory limit: 128 MB.
 Teachpack: convert.rkt.
underlay: this function is not defined
 >

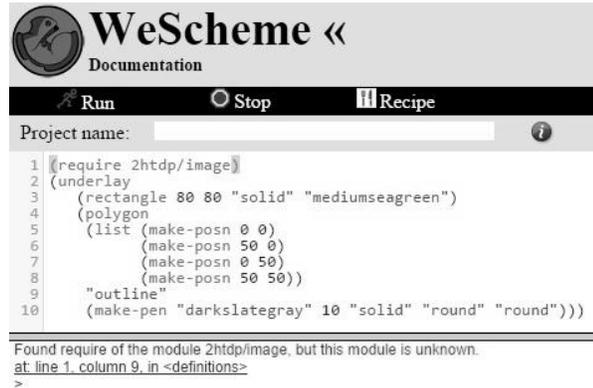
Welcome to **DrRacket**, version 6.0.1 [3m].
 Language: Beginning Student [custom]; memory limit: 128 MB.
 Teachpack: convert.rkt.

 >

En el caso del ambiente de programación *WeScheme* el modulo es totalmente desconocido y por lo tanto ocurre un error.



WeScheme «
 Documentation
 Run Stop Recipe
 Project name:
 1 (underlay
 2 (rectangle 80 80 "solid" "mediumseagreen")
 3 (polygon
 4 (list (make-posn 0 0)
 5 (make-posn 50 0)
 6 (make-posn 0 50)
 7 (make-posn 50 50))
 8 "outline"
 9 (make-pen "darkslategray" 10 "solid" "round" "round")))
make-pen : this variable is not defined
 at: line 9, column 4, in <definitions>
 at: line 3, column 3, in <definitions>
 at: line 1, column 0, in <definitions>
 >



WeScheme «
 Documentation
 Run Stop Recipe
 Project name:
 1 (require 2htdp/image)
 2 (underlay
 3 (rectangle 80 80 "solid" "mediumseagreen")
 4 (polygon
 5 (list (make-posn 0 0)
 6 (make-posn 50 0)
 7 (make-posn 0 50)
 8 (make-posn 50 50))
 9 "outline"
 10 (make-pen "darkslategray" 10 "solid" "round" "round")))
 Found require of the module 2htdp/image, but this module is unknown.
 at: line 1, column 9, in <definitions>
 >

Se debe utilizar el ambiente *WeScheme* sin preocupación a pesar del ejemplo en las imágenes pasadas, ya que solo se enfocara en la enseñanza y por lo tanto no se abarca todo el lenguaje.

3.3 Diseño de Programas

Definición de constantes.

La definición de una constante se forma abriendo un paréntesis escribiendo “*define*” después el nombre de la constante y un espacio para dar el tipo de valor que llevara este nombre después de ser declarado.

```
(define WIDTH 400)
```



Nombre de la
constante

Una de las facilidades que se tiene de la definición de constantes, es el uso de imágenes en el desarrollo de un programa, ya que poner imágenes en un extenso código es complicado y se facilita más usando el nombre de la constante.

```
(define cat )
```

```
1 (require 2htdp/image)
2
3
4 (define cat )
5 cat
6
```

Lenguaje: Estudiante Principiante; memory limit: 128 MB.

```

>
```

Las norma para la evaluación de una definición de constante es cuando la evaluación se encuentra con una constante con nombre, el nombre de constante, simplemente se evalúa con el valor que se asoció.

Funciones

El manejo de funciones es importante durante el aprendizaje de *Racket* y también para la enseñanza de las matemáticas bajo este ambiente de programación, ya que se puede idear una serie de ejercicios para crear expresiones que se manejen de forma general para resolver el mismo problema pero con diferentes valores.

Ejemplos:

Cuál es el resultado de $x^2 - 2$

En el lenguaje de *Racket* se expresa de la siguiente manera:

```
(- (sqr x) 2)
```

Existe un error en la siguiente expresión si se quiere usar directamente en la ventana de interacciones ya que la x no es un valor numérico y hay que definirla.

```
(- (sqr x) 2)
x: this variable is not defined
at: line 1, column 3, in <definitions>
at: line 1, column 0, in <definitions>
```

La manera correcta de expresar una función es la siguiente:

```
(define (resolver x)
  (- (sqr x) 2))
```

resolver: es el nombre de la función

x: es la variable

Pero existe una forma fácil y sólida para crear funciones, ya que existe una receta de diseño que no precisamente es como una receta de cocina, se menciona paso a paso en el libro "*How to Design*

Programs” pero tomare la receta de diseño recientemente proporcionada por el curso de *Gregor Kiczales* en *Coursera* que a mi parecer me parece más completa.

Firma

La firma se forma primero por los datos de entrada y después de ello se pone guion medio y “mayor que” para formar una flecha después de escribimos el tipo de datos que produce nuestra función.

```
imagen -> imagen
```

Propósito

Un comentario breve de qué calcula el programa; para la mayoría de los programas que se verán, una o dos líneas serán suficientes; en la medida que se desarrollen programas cada vez mayores, se requerirá añadir más información con objeto de explicar el propósito del programa.

Borrador o stub

Primero debemos tener la firma, ya que gracias a ella es más fácil crear el borrador porque se tiene los tipos de datos de entrada y de salida. Para crear un borrador se escribe lo que pareciera una función pero tan solo se quiere probar cuando no existe expresión en la función y solo contiene el tipo de dato predeterminado por tanto no existe variable donde probar algún tipo de valor y producirá un error mencionando que los datos no concuerdan.

;; Borrador

```
; (define (adds? s) " ")
```

Pruebas

Con objeto de comprender mejor qué debe calcular el programa, se estructuran ejemplos de entradas y se determina cuáles deben ser sus salidas.

;; Pruebas

```
(check-expect (adds? "cat") "cat?")  
(check-expect (adds? "what?") "what?")
```

Formato:

Es la plantilla de cómo va estar formada tu función.

Cuerpo de la función:

En esta parte se completa la creación final de la función.

Se usa la plantilla del formato se creara el cuerpo de la función pero también es necesario tener las expresiones que usan las pruebas hacer completar con la construcción de misma se toma la expresión de las pruebas identificando que parte de esa expresión será la parte que varía para que esa expresión se convierta en una expresión general.

3.4 Programación

Se observa desde la perspectiva de la codificación en *WeScheme* y *Racket* que un programa sólo son funciones y definiciones de constantes. Usualmente, la función que más destacaría, es *main*, ya que es la función principal y tiene como papel coordinar (invocar) a las distintas funciones para que el programa lleve a cabo su cometido exitosamente. Pero existen dos clases distintas de programas.

3.4.1 Programación *batch*

Un programa *batch* consume todas sus entradas a la vez y calcula sus resultados. Su función principal se compone de funciones auxiliares, que se refieren a funciones auxiliares adicionales. Cuando se lanza un programa *batch*, el sistema operativo llama a las entradas de la función *main* y espera la salida del programa.

Como se ha mencionado, un programa *batch* consume todas sus entradas a la vez y calcula el resultado de sus salidas. Su función principal puede esperar los propios argumentos o los nombres de los archivos desde el cual recupera las entradas y regresa la salida o ser colocado en un archivo.

Los programas son incluso más útiles si se puede recuperar la entrada de algún archivo y entrega la salida de algún otro archivo. El nombre del programa *batch* se originó de los primeros días de las computación cuando un programa lee un archivo entero (o varios archivos) y coloca el resultado en algunos otros archivo(s), sin alguna intervención después de ejecutarlos. Conceptualmente, se puede pensar de un programa como la lectura de un archivo completo de una sola vez y producir el resultado de todos los archivo(s) de una vez.

Creamos programas *batch* basados en archivos con el *teachpack* "*batch-io*", que agregan dos funciones a nuestro vocabulario (entre otros):

read-file, el cual lee el contenido de un archivo completo como una cadena

write-file, el cual crea un archivo de una cadena dada

Las funciones escriben cadenas a archivos y leen cadenas de los archivos

```
> (write-file "sample.dat" "212")
"sample.dat"
> (read-file "sample.dat")
"212"
```

Después de que se completa la primera interacción, el archivo llamado "*sample.dat*" contiene el texto
212

El resultado de *write-file* es un reconocimiento de que se ha colocado la cadena en el archivo. Si ya existe el archivo, reemplaza su contenido con la cadena dada; de lo contrario, se crea un archivo y hace que la cadena dada sea su contenido. La segunda interacción, (*read-file "sample.dat"*), produce "212" porque el contenido de "*sample.dat*" es una cadena.

Por razones históricas y pragmáticas, *write-file* también acepta *'stdout*, un tipo especial de razón, como primer argumento. A continuación, se muestra el contenido del archivo resultante en el área de interacciones, por ejemplo:

```
> (write-file 'stdout "212")
212
'stdout
```

Por analogía, el *read-file* acepta *'stdin* in lugar de un nombre de archivo y luego lee la entrada del área de interacciones.

Se ilustra con la creación de un programa *batch* con un simple ejemplo. Se crea un programa que convierte una temperatura medida en un termómetro de Fahrenheit a temperatura Celsius. Se tiene la siguiente expresión que realiza la conversión.

$$c = \frac{5}{9} \cdot (f - 32)$$

Naturalmente, en la fórmula *f* es la temperatura en grados Fahrenheit y *C* es la temperatura en Celsius. Para los programadores o matemáticos se escribe *c(f)* del lado izquierdo de la ecuación para recordar que *f* es un valor determinado y *c* se calcula a partir de *f*.

Se traduce a una expresión reconocida por *Racket* en la opción lenguaje principiante.

```
(define (f2c f)
  (* 5/9 (- f 32)))
```

Se recuerda que *5/9* es un número, una fracción racional para ser preciso y lo más importante, que *c* depende del valor de *f* dado, ya que lo expresa la notación de la función.

```
> (f2c 32)
0
> (f2c 212)
100
> (f2c -40)
-40
```

Si se utiliza la misma función como parte de un programa que lea la temperatura en Fahrenheit de un archivo, convierta el valor en una temperatura en Celsius, y crea un archivo que contenga el resultado.

Una vez que se tiene la fórmula de conversión, la creación de la función principal *f2c* dependerá de funciones primitivas existentes:

```
(define (convert in out)
  (write-file out
    (number->string
      (f2c
        (string->number
          (read-file in))))))
```

Se ha llamado a la función principal *convert*, consume dos nombres: en el archivo donde se tiene la temperatura en Fahrenheit y se quiere un resultado en Celsius. Se compone de cinco funciones que calculan el resultado del *convert*. Se observa cuidadosamente para entender cómo funciona:

`(read-file in)` se recupera el contenido del archivo llamado como una cadena

`string->number` se convierte la cadena en número

`f2c` se interpreta el número como una temperatura Fahrenheit y convierte en un valor que representa la temperatura en Celsius.

`number->string` consume el valor dado en Celsius y lo convierte en una cadena

`(write-file out ...)` escribe la cadena en el archivo llamado *out*

Se debe tener en cuenta que los programas logran un propósito en el mundo real, a su vez los ejercicios de álgebra ilustran problemas matemáticos de aplicación general.

Una forma de crear archivos es con la función *write-file*, ejecutando *convert*:

```
> (write-file "sample.dat" "212")
"sample.dat"
> (convert "sample.dat" 'stdout)
100
'stdout
> (convert "sample.dat" "out.dat")
"out.dat"
> (read-file "out.dat")
"100"
```

La primera interacción, se usa `'stdout` para observar las salidas en la zona de interacciones de *DrRacket*. Para la segunda de *convert* se le da el nombre de *out.dat*. Era de esperar que la llamada a la conversión devuelva una cadena, al escribir el archivo se sabe que se depositó una temperatura Fahrenheit en el archivo. Aquí leemos el contenido del archivo con *read-file*.

```
(define (convert in out)
```

```

(write-file out
  (number->string
    (f2c
      (string->number
        (read-file in))))))

(define (f2c f)
  (* 5/9 (- f 32)))

(convert "sample.dat" "out.dat")

```

3.4.2 Programación interactiva

Un programa interactivo llama a la entrada de un evento, y se crean programas interactivos como programas orientados a eventos. La función principal de tal programa usa una expresión para describir las funciones a llamar de los diferentes tipos de eventos. Cuando se ejecuta el programa interactivo, donde la función principal informa al sistema operativo de su descripción, inmediatamente en cuanto ocurren los eventos de entrada, el sistema operativo llama a las funciones de coincidencia. Del mismo modo, el sistema operativo sabe su descripción, cuándo y cómo se presentan los resultados de las funciones llamadas, así como su salida.

Las personas interactúan con aplicaciones de escritorio a través de un teclado o un ratón, generando eventos de entrada tales como presionar el teclado o clics en el mouse. Además, los programas interactivos, también pueden reaccionar a los eventos generados por la computadora, por ejemplo, ciclos de reloj o llegada de un mensaje de alguna computadora.

La mayoría de las personas ya no usan principalmente sus computadoras de escritorio para ejecutar aplicaciones, ya que hoy se utilizan los teléfonos celulares, tabletas y pantallas de control en sus automóviles. La gente usará computadoras *wearables* en forma de gafas inteligentes, ropa y equipo

deportivo. En un futuro no muy lejano, la gente puede vivir con computadoras bio-integradas para interactuar con funciones del cuerpo.

Una computadora es la parte física llamada hardware y es una parte inútil hasta que se combina con la parte del software, que es un conjunto de programas que trabajan juntos para un solo fin. La primera pieza del software que se instala es el sistema operativo que se encarga de gestionar el equipo para el usuario, incluyendo los dispositivos conectados, como: monitor, teclado, mouse, sonido, etc. Un ejemplo sencillo es cuando un usuario al pulsar una tecla en el teclado, el sistema operativo ejecuta una función que procesa los movimientos. Se dice que la clave es un evento, y la función es un controlador de eventos. El sistema operativo ejecuta un controlador de eventos para el avance del reloj, las acciones del mouse, etc. Después del trabajo del controlador de eventos, el sistema operativo puede tener cambios en la imagen de pantalla, sonidos o impresión de un documento. Se realizan las tareas gracias a la ejecución de funciones que traducen datos del sistema operativo en sonidos, imágenes y acciones en una impresora.

Naturalmente los diferentes programas tienen diferentes necesidades. Un programa puede interpretar como señales de pulsaciones de teclas para controlar un reactor nuclear, otro para un procesador de textos. Para realizar diferentes tareas en una computadora se necesitan diferentes programas que usen diferentes controladores de eventos.

El diseño de un programa interactivo requiere una forma de designar alguna función como el encargado de los eventos de teclado, la función para tratar con el tic de reloj, una tercera función para representar algunos datos como imágenes. La tarea de la función principal de un programa interactivo es comunicar las diferentes llamadas al sistema operativo.

En un ambiente de programación como *WeScheme*, una expresión *big-bang*, consiste de una sub expresión requerida y, por ejemplo, una cláusula. La sub expresión se evalúa como el estado inicial del programa, y la cláusula requiere *WeScheme* para representar el estado actual del programa.

La idea se entiende paso a paso, a partir de la definición de la función:

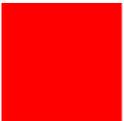
```
(define (number->square s)
  (square s "solid" "red"))
```

La función consume un número positivo y produce un cuadrado sólido rojo de un cierto tamaño. Si se hace un clic en *Run*, se experimenta con la función de la siguiente manera:

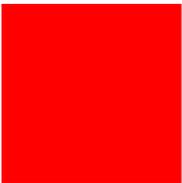
```
> (number->square 30)
```



```
> (number->square 60)
```



```
> (number->square 90)
```



En relación a la siguiente expresión *big-bang* en el área de interacciones:

```
> (big-bang 100 [to-draw number->square])
```

Dibuja una ventana y muestra un cuadrado rojo de cien por cien píxeles. Además en el área de interacciones de *WeScheme* no muestra un mensaje; es como si el programa siguiera funcionando y en verdad es así. Se detiene el programa haciendo clic en cerrar de la ventana donde contiene el cuadrado.

```
> (big-bang 100 [to-draw number->square])
```

Se tiene una interesante expresión *big-bang*:

```
> (big-bang 100
    [to-draw number->square]
    [on-tick sub1]
    [stop-when zero?])
```

Ahora en la expresión *big-bang* se añaden dos cláusulas adicionales al anterior: *on-tick* y la cláusula *stop-when* que se relaciona con los tics del reloj y de parada del programa. Se lee que a partir del 100 como estado inicial:

- 1.- Cada vez que el reloj avanza, restar 1 del estado actual
- 2.- A continuación comprueba si es cero, si es *true* se debe detener
- 3.- Cada vez que un manejador de eventos finaliza con su trabajo, emplea *number->square* para representar el estado como una imagen.

CAPITULO 4

Desarrollo e integración de recursos

4.1 Metodología de integración

Una de las formas más efectivas de saber si existe el dominio acerca de cualquier tema de matemáticas es a través de una evaluación, para los profesores es de gran ayuda, ya que se puede observar si el alumno está capacitado o su dominio de los temas. En el caso del alumno se puede dar cuenta de los conocimientos que carece para resolver algunos de los ejercicios, ya que lo conduce a encontrar ciertas dudas que debe resolver, pero a veces la efectividad se encuentra en el método con el cual se brindan los conocimientos.

Al emplear materiales de *Bootstrap:1* el estudiante, de forma autodidacta o bajo la guía del profesor, puede relacionar elementos de videojuegos con conceptos matemáticos como es el espacio cartesiano, lo que a su vez permite profundizar en esos elementos de videojuegos; lo que a su vez requiere que aprenda a diseñar funciones, ampliando el manejo de conceptos algebraicos, geométricos y trigonométricos. Esto lo conduce de forma natural a manejar animaciones elementales, demandando a su vez que estructure el diseño de funciones en el diseño de *mundos*, lo que a su vez requiera que modele aspectos de la realidad física con conceptos matemáticos más avanzados.

Ese ir y venir del manejo de expresiones, diseño de funciones y *mundos*, con el manejo creciente e integral de conceptos matemáticos, pueden apoyarse en recursos web como las páginas de la División de Ciencias Básicas de la Facultad de Ingeniería al inicio de este trabajo, entre otros. En esa perspectiva se pueden referir estos recursos en un orden y con un sentido que aumenten su eficacia y eficiencia, éstas con base en observaciones como las realizadas cuando se les examino brevemente. Lo mismo puede suceder en una perspectiva iterativa con los demás recursos web examinados, integrándolos de forma creciente, con mayor eficacia y eficiencia.

4.2 Ejemplos de integración

Las matemáticas no sólo expresan sus soluciones mediante cálculos, sino que desarrollan todo un lenguaje para condicionar el contexto en el cual se puede aplicar una solución y desarrollan la idea

fundamental de construir soluciones complejas mediante la combinación de soluciones a problemas más simples. Un ejemplo es hallar las raíces de una expresión cuadrática x^2+bx+c donde se puede considerar que existen dos números x_1 y x_2 que si suman su resultado es “b” y la multiplicación es “c”, esto se resuelve teniendo conocimientos de multiplicar y sumar que todos sabemos realizar muy bien y estamos enterados de sus condiciones y propiedades.

El alcance al cual se quiere llegar es ser capaz de diseñar programas simples. La comprensión de cómo funcionan tales programas ayuda a entender el funcionamiento desde teléfonos inteligentes, hasta sitios web, sistemas bancarios, juegos, películas, redes sociales y muchos otros sistemas computacionales. El ser capaz de diseñar programas apoya al trabajo como científico, ingeniero, y artista, entre otros: la programación permite organizar, almacenar, analizar y visualizar información, crear animaciones, música y comunidades en línea, así como manejar dispositivos de control de nuestro entorno; además del desarrollo de modelos computacionales y simulaciones, y muchos más.

Pero, en esta tesis este lenguaje se usa con el objetivo de introducir a los estudiantes a las matemáticas y a un método sistemático para el diseño de programas.

Por ejemplo, si inicialmente se requiere manejar expresiones aritméticas, geométricas y trigonométricas, abarcando las que se deben manejar como antecedentes de secundaria y preparatoria para aprender conceptos matemáticos de asignaturas del primer semestre en las carreras de ingeniería de la Facultad. Una posible secuencia podría ser emplear recursos web como sigue:

Se toman ejercicios de los diferentes recursos vistos en el capítulo 2

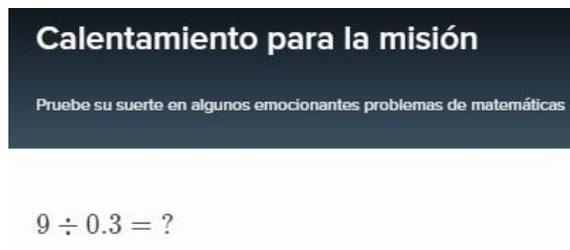
Ejercicios muy elementales de aritmética⁸⁵ (Evaluación de Aritmética, 2014) y trigonometría⁸⁶ (Evaluaciones de Álgebra I, 2014) tomados de *KhanAcademy*, así como ejercicios de polinomios que

⁸⁵ KhanAcademy, Evaluación de Aritmética <https://es.khanacademy.org/mission/arithmetical> (consultado el 29 de septiembre de 2014)

⁸⁶ KhanAcademy, Evaluaciones de Álgebra I, <https://es.khanacademy.org/mission/algebra> (consultado el 29 de septiembre de 2014)

se encuentran en la evaluación de operaciones algebraicas⁸⁷ (Evaluacion, 2014) de la página de la División de Ciencias Básicas⁸⁸.

Ejercicios de aritmética con expresiones simples⁸⁹ (Evaluación de Aritmética, 2014).



(/ 9 0.3)

30

Otro ejemplo:

$3 - 1 = ?$

(- 3 1)

2

Se ejemplifica mostrando los pasos de la expresión dada y se desarrolla hasta llegar al valor esperado. Esto se muestra para indicar que se realiza de izquierda a derecha y se calculan primero los paréntesis más internos hasta llegar al valor.

(- (* 9 9) (- (/ 8 2) 3))

(- 81 (- (/ 8 2) 3))

(- 81 (- 4 3))

(- 81 1)

80

⁸⁷ Cuestionario de Álgebra, http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/operaciones_algebraicas.html (consultado el 29 de septiembre de 2014)

⁸⁸ DCB, División de Ciencias Básicas, <http://dcb.fi-c.unam.mx/> (consultado el 29 de septiembre de 2014)

⁸⁹ KhanAcademy, Evaluación de Aritmética <https://es.khanacademy.org/mission/arithmic> (consultado el 29 de septiembre de 2014)

`(+ (* 9 9) (- (- 8 3) 3))`

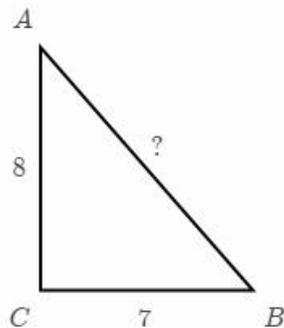
`(+ 81 (- 5 3))`

`(+ 81 2)`

83

Se retoma un ejercicio visto en la imagen siguiente de *KhanAcademy*⁹⁰ (Álgebra I, 2014)

En el triángulo rectángulo que se muestra, $AC = 8$ y $BC = 7$. ¿Cuánto vale AB ?



Se debe utilizar el teorema de Pitágoras, ya que se trata de un triángulo rectángulo.

$$c^2 = a^2 + b^2$$

En el ejercicio se cuenta con los valores de sus lados donde $a = AC = 8$ y $b = BC = 7$. Se debe encontrar el lado $c = AB = ?$

a) Mostrar la expresión en lenguaje *Racket*

`(sqrt (+ (sqr 8) (sqr 7)))`

b) ¿Cuál es el resultado obtenido en el ambiente de programación?

10.63014581273465

Se cuenta con otro ejemplo basado en el curso *Introduction to Systematic Program Design - Part 1*⁹¹

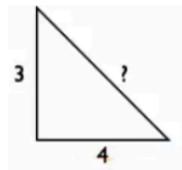
⁹⁰ Álgebra I, Teorema de Pitágoras, <https://es.khanacademy.org/mission/algebra> (consultado el 30 septiembre de 2014)

⁹¹ Coursera, Curso Introduction to Systematic Program Design – Part 1, <https://class.coursera.org/programdesign-002> (consultado el 30 de septiembre de 2014)

(curso Kiczales, 2014) impartido por Gregor Kiczales en *Coursera*

Se observa de manera clara la resolución del problema con el uso del Teorema de Pitágoras.

Asumir que los dos lados menores de un triángulo rectángulo tienen longitud 3 y 4. ¿Cuál es la longitud del lado mayor (hipotenusa)?



$H = \text{Hipotenusa}$

$C.O = \text{Cateto Opuesto}$

$C.A = \text{Cateto Adyacente}$

$$H = \sqrt{C.O^2 + C.A^2} \quad H = ? \quad H = \sqrt{3^2 + 4^2}$$

Escribir una expresión del Lenguaje Estudiante Principiante, *LEP* o *BSL* (*Beginning Student Language*) que produzca el valor.

Donde en *BSL* de *Racket*.

`sqrt`: es la raíz de un número.

`sqr`: es el cuadrado de un número.

La respuesta es la siguiente expresión.

```
(sqrt (+ (sqr 3) (sqr 4)))
```

Para este caso el valor resultante de la expresión de arriba en código *Racket*, significa la longitud de la hipotenusa del triángulo rectángulo. 3 y 4 son los valores de los catetos para este problema en particular.

Ahora este ejercicio se puede extender más si se pide que se desarrolle paso a paso la expresión BSL hasta llegar al valor de la hipotenusa. El desarrollo se realiza de la siguiente manera:

```
(sqrt (+ (sqr 3) (sqr 4)))  
(sqrt (+ 9 (sqr 4)))  
(sqrt (+ 9 16))  
(sqrt 25)  
5
```

Ejemplos de polinomios, donde algunos ya fueron mencionados en el Capítulo 2. Recursos de la División de Ciencias Básicas⁹² (operaciones algebraicas, 2014).

Ejemplo 1:

Operaciones algebraicas

1.- Después de reducir la expresión $(-3x - (x+2)(x-4)) + (x+3)(x-1)$ se obtiene

- $x^2 - 3x - 11$
- $x + 5$
- $- 3x - 11$
- $- 3x + 5$

Se utiliza la declaración de una variable constante para poder sustituir x por un valor. El valor será 2 y se utilizara la misma para probar todos los ejemplos.

```
(define x 2)
```

⁹² Cuestionario de Álgebra, http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/operaciones_algebraicas.html (consultado el 29 de septiembre de 2014)

Se utiliza `check-expect` para comprobar que la primera expresión tiene el mismo valor que la expresión esperada.

```
(check-expect expression expected-expression)
```

Si la prueba es satisfactoria, no se debe tener errores y no tiene un resultado visible, en el caso de *WeScheme*⁹³ (Ejercicio en *WeScheme*, 2014).

```
(check-expect (+ (- (* -3 x) (* (+ 2 x) (- x 4)))) (* (+ x 3) (- x 1)))
              (+ x 5))
```

Pasa la prueba y la respuesta correcta es: `(+ x 5)`

Ejemplo 2:

2.- Al desarrollar la expresión $(x^2 + 6x + 3)(x - 5)$ se obtiene

- $x^3 + x^2 - 27x - 15$
- $x^3 - 11x^2 + 33x + 15$
- $x^3 - x^2 + 33x - 15$
- $x^3 + 7x - 27x - 15$

```
(define x 2)
```

```
(check-expect (* (+ (sqr x) (* 6 x) 3) (- x 5))
              (+ (expt x 3) (sqr x) (* -27 x) -15))
```

Pasa la prueba y la respuesta correcta es: `(+ (expt x 3) (sqr x) (* -27 x) -15)`

⁹³ Ejercicios creados en *WeScheme*, <http://www.wescheme.org/openEditor?publicId=JzP3nDmUv> (consultado el 30 de septiembre de 2014)

Ejemplo 3:

3.- Al efectuar la operación $1 + \frac{1}{x-3} + \frac{3}{x(x-3)}$ se obtiene

- $\frac{x^2 - 2x + 3}{x(x-3)}$
- $\frac{2}{x}$
- $\frac{x+1}{x}$
- $\frac{-3x^2 + x - 8}{x(x-3)}$

(check-expect (+ 1 (/ 1 (- x 3)) (/ 3 (* x (- x 3))))
(/ (+ (sqrt x) (* -2 x) 3) (* x (- x 3))))

Pasa la prueba y la respuesta correcta es: (/ (+ (sqrt x) (* -2 x) 3) (* x (- x 3)))

Ejemplo 4:

4.- Al efectuar la operación $\frac{x}{x+1} + \frac{2}{x+1} - \frac{3}{x+1}$ se obtiene

- $\frac{x-1}{x+1}$
- $x-3$
- $\frac{(x+3)(x-1)}{x(x+1)}$
- $\frac{2x-3}{x}$

(check-expect (+ (/ x (+ x 1)) (/ 2 (+ x 1)) (/ -3 (+ x 1)))
(/ (- x 1) (+ x 1)))

Pasa la prueba y la respuesta correcta es: (/ (- x 1) (+ x 1))

En el caso de una respuesta falsa

```
(check-expect (+ (/ x (+ x 1)) (/ 2 (+ x 1)) (/ -3 (+ x 1)))  
              (- x 3))
```

La respuesta tiene un error: check-expect: actual value 1/3 differs from -1, the expected value.

[at: line 3, column 0, in <definitions>](#)

Las dos expresiones no son iguales.

Ejemplo 5:

5.- Al simplificar la expresión $(x^2 + (x+2)(x-3)) - (2x+1)(x-2)$ se obtiene

- $-4x - 8$
- $-2x - 4$
- $2x - 4$
- $8x - 4$

```
(check-expect (- (+ (sqr x) (* (+ x 2) (- x 3))) (* (+ (* 2 x) 1) (- x 2)))  
              (- (* 2 x) 4))
```

Pasa la prueba y la respuesta correcta es: $(- (* 2 x) 4)$

Se muestra otro ejemplo referido a Álgebra:

Aquí se observa una forma sencilla de poder jugar con polinomios y el lenguaje *Racket*.

Convertir las expresiones algebraicas en expresiones válidas para el lenguaje *Racket*.

$$\begin{aligned}
 & -x^3 + 4x^2 - 6 \\
 & x^2 + 3xy - y^2 \\
 & -5x^2y^2 - 8 + 2x^4
 \end{aligned}$$

Como se ha mencionado antes acerca de la forma en que se opera ya se tiene noción de que primero se abre un paréntesis después se inicia con el operador y después lo elementos destinados a ser operados:

Para la expresión $-x^3 + 4x^2 - 6$ se realiza de la siguiente forma:

Primeramente iniciamos guiándonos por la siguiente forma:

(operador argumento1, argumento2, argumentoN)

Resolvemos por partes

Para el caso de $4x^2$

(* 4 (sqr x))

Para el caso de x^3 :

(* (sqr x) x)

Para el caso de $-x^3 + 4x^2$:

(- (* 4 (sqr x)) (* (sqr x) x))

Teniendo las demás partes de la expresión tan solo falta restarle 6:

$$-x^3 + 4x^2 - 6$$

```
(- (- (* 4 (sqr x)) (* (sqr x) x)) 6)
```

De la misma forma se resuelven los siguientes ejercicios

Ejemplos con imágenes.

En el siguiente ejemplo aplicamos la función definida *triangle/sss* que construye un triángulo con los tres de sus lados conocidos para este caso es un triángulo equilátero ya que todos sus lados son iguales.

Se utiliza también el operador *rotate* para rotar la imagen dándole un número arbitrario de grados, hasta donde se desea ubicar la imagen.

Ejemplo 1:

Se menciona sus características y dibujar un triángulo equilátero con ayuda del lenguaje *Racket*.

```
(rotate 60 (triangle/sss 60 60 60 "solid" "seagreen"))
```



Si no se emplea *rotate* quedaría el resultado de la siguiente manera.

```
(triangle/sss 60 60 60 "solid" "seagreen")
```



Si se desea el triángulo delineado

```
(rotate 60 (triangle/sss 60 60 60 "outline" "seagreen"))
```



Ejemplo 2:

Mencionar características de un triángulo rectángulo y dibujarlo usando el lenguaje *Racket*.

```
(triangle/ass "ángulo-conocido" "lado-cateto-a" "lado-cateto-b" "modo" "color")
```

El modo tiene las siguientes opciones `outline` o `solid`

Solución con código *Racket* o *WeScheme*.

```
(triangle/ass 90 50 80 "solid" "blue")
```



Ejemplo 3:

Mencionar las características que tiene un triángulo isósceles y dibujarlo usando el lenguaje *Racket*.

```
(triangle/sss "lado-a" "lado-b" "lado-c" "modo" "color")
```

El modo en que se tiene las siguientes opciones *outline* o *solid* que se escriben de dos formas.

```
(rotate 69 (triangle/sss 70 50 70 "solid" "seagreen"))
```



Ejercicios donde se utiliza la receta de diseño para crear funciones.

Ejercicio1:

Comenzamos a plantear un tema que es la conversión de grados a radianes para esto necesitamos entender lo siguiente para poder realizar un ejemplo con *Racket* y a su vez aprender Matemáticas:

Cuando θ es igual a 180° , entonces el arco al que se había nombrado L que interseca sobre un círculo de radio r , es una semicircunferencia tal que $L = \pi * r$, donde π es aproximadamente 3.1416. De acuerdo con lo anterior, si se aplica la ecuación para obtener la medida de un ángulo en radianes, se tiene:

$$180^\circ = \frac{L}{r} \text{ rad} = \frac{\pi r}{r} \text{ rad} = \pi \text{ rad}$$
$$180^\circ = \pi \text{ rad}$$

De esta última igualdad se utiliza para convertir el valor de un ángulo de grados a radianes y viceversa. Así se obtiene lo siguiente.

$1^\circ = \frac{\pi}{180} \text{ rad}$, donde $\frac{\pi}{180}$ equivale aproximadamente a 0.017453

Así que: $1^\circ = 0.017453 \text{ rad}$

Y también $1 \text{ rad} = \frac{180^\circ}{\pi}$, donde $\frac{180^\circ}{\pi}$ equivale aproximadamente a 57.296°

Así que: $1 \text{ rad} = 57.296^\circ$

Estos dos tipos de medidas tienen su equivalencia que es la siguiente:

$$1 \text{ rad} = \frac{180}{\pi} \text{ grados} \rightarrow 1 \text{ rad} = 57.2958 \text{ grados}$$
$$1 \text{ grado} = \frac{\pi}{180} \text{ radianes} \rightarrow 1 \text{ grado} = 0.01745 \text{ rad}$$

Usando *WeScheme* se realiza las funciones de conversiones en *Racket*, y quedan de la siguiente forma.

Usando la receta de diseño de programas⁹⁴ (Receta de diseño, 2014) se resuelve el problema de la siguiente forma.

Se mencionan los tipos de datos que entran y los que se generan y también el nombre de la función.

Conversión de grados a radianes.

; Función gradARad

⁹⁴ How to design programs 2da edición, Receta de diseño, http://www.ccs.neu.edu/home/matthias/HtDP2e/part_one.html#%28part._ch~3ahtdp%29 (consultado el 30 de septiembre de 2014)

; Firma

Number -> Number

; Borrador (stub):

```
(define (gradARad b) 0)
```

; Formato

```
; (define (gradARad b)
; ( ... b ...))
```

; Pruebas

```
(check-within (gradARad 2) (* 2 (/ 1 (/ 180 pi))))
(check-within (gradARad 180) (* 180 (/ 1 (/ 180 pi))))
```

; Cuerpo de la función

```
(define (gradARad b)
  (* b (/ 1 (/ 180 pi))))
```

Ejemplo 2:

Teorema 1. La suma de los tres ángulos internos de todo triángulo es igual a 180° o en radianes a π

$$\alpha + \beta + \gamma = 180^\circ$$
$$(\text{= (+ } \alpha \beta \gamma) 180)$$

Con lo antes visto realizar una función que se nombrara *sumaAngT* que recibe dos ángulos conocidos y produce el ángulo que sumado con los dos ángulos que recibe da por resultado 180°.

; Función sumaAngT

; Firma:

; Number Number -> Number

; Propósito:

; Se genera en un número que sumado con los dos números que recibe
; sea igual a 180.

; Borrador (stub):

```
(define (sumaAngT a1 a2) 0)
```

; Formato:

```
; (define (sumaAngT a1 a2)  
; (... a1 ... .. a2 ...))
```

; Pruebas:

```
(check-expect (sumaAngT 30 45) (- 180 (+ 30 45)))  
(check-expect (sumaAngT 90 30) (- 180 (+ 90 30)))
```

; Definición de la función:

```
(define (sumaAngT a1 a2)  
  (- 180 (+ a1 a2)))
```

Diseño de un mundo sencillo

Se dispone de una API de programación de mundos e imágenes.

El siguiente ejemplo nos ilustra sobre el uso de *big-bang* como de sus cláusulas que interactúan con dispositivos periféricos con el programa creado.



Se realiza una descripción del código que aparece en la imagen anterior.

Se menciona que un mundo es un mundo

```
;; El mundo es un número
```

El contrato

Indica el nombre de la función "maneja-tecla". Se menciona los datos de entrada, un mundo que es un valor numérico y una tecla alfanumérica. La salida es una imagen.

```
;; maneja-tecla: mundo tecla -> imagen
```

Regresa *#t* si *v* es un carácter y *#f* si no lo es.

Se define la función *maneja-tecla* que tiene dos entradas *m t*, donde *m* es un mundo que representa un número y *t* es un evento de teclado.

```
(maneja-tecla m t)
```

En el caso del dato de entrada *t* se debe explicar detalladamente sobre los eventos del teclado, ya que un evento *key* representa eventos del teclado, pulsa o deja de presionar teclas. Se utiliza *char?* para indicar que el usuario ha llegado a un teclado alfanumérico. Los símbolos utilizados son *'left*, *'right*, *'up*, *'down*, *'release* y denota teclas de flecha o eventos especiales, como la liberación de teclas en el teclado.

En el caso de *char?* se tiene:

```
(char? v) → boolean?
```

v : any/c

Si *v* es *#t*

El ejemplo tiene un condicional que contiene cláusulas donde aparece *key=?* y realiza una comparación de teclas alfanuméricas

```
(cond [(key=? t "up")
      (add1 m)]
      [(key=? t "down")
      (sub1 m)]
      [else
      m])
```

Si en la comparación que se realiza en la primera clausula es igual a *"up"* se sustituye el valor de *m* en la siguiente expresión (*add1 m*), y produce un valor numérico.

```
[(key=? t "up") (add1 m)]
```

Si en la comparación que se realiza en la segunda clausula es igual a *"down"*, se sustituye el valor de *m* en la siguiente expresión `(sub1 m)`, y produce un valor numérico.

```
[(key=? t "down") (sub1 m)]
```

Si no se cumple ninguna de las comparaciones de las dos clausulas, el valor que entra será el mismo que sale. Significa que no se oprime ninguna de las teclas mencionadas en cada clausula.

```
[else m]
```

EL *big-bang* consume un mundo inicialmente, así como múltiples gestores para configurarlo, se describe en la siguiente sección:

Si se continúa con la parte de *big-bang*, sus argumentos muestran características importantes utilizadas para resolver problemas de *mundos*, ya que se observa como parte de sus argumentos, un valor inicial cero, así como la función *on-key* que informa a *WeScheme* que debe llamar a la función *maneja-tecla* para actualizar el mundo y el evento de las teclas al oprimir el teclado. Lo que resulta de la llamada se convierte en el mundo actual.

```
(big-bang 0
         (on-key maneja-tecla))
```

```
(define (maneja-tecla m t)
  (cond [(key=? t "up")
        (add1 m)]
        [(key=? t "down")
        (sub1 m)]
        [else
         m]))
```

```
(big-bang 0
  (on-key maneja-tecla))
```

Un segundo ejemplo ilustrativo es el siguiente:

```
;; El mundo es un numero.
;; dibuja: mundo -> imagen
(define (dibuja mundo)
  (circle mundo "solid" "blue"))
```

```
(big-bang 20
  (to-draw dibuja))
```



Conclusión

Se puede concluir que disponer de recursos para reducir carencias relacionadas con antecedentes de Matemáticas, que hoy según las estadísticas es un problema común y casi universal de los estudiantes, a través de involucrar principios de programación conjuntamente con la enseñanza de temas de Matemáticas, es un nuevo punto de partida tanto del aprendizaje de las matemáticas, como de la programación, orientado a una nueva manera de aprender más allá de sólo resolver ejercicios.

El diseño de programas obliga a ser cuidadoso con la definición de datos, definir firmas, propósitos, formatos y pruebas. Lo que da un sentido adicional a apoyarse en recursos en línea, los que por su parte son cada vez más populares, accesibles, y de mejor calidad.

Sin olvidar que todo método, recurso y forma de aprendizaje será tan efectivo como sean la curiosidad, el interés y la motivación del estudiante. Los cuales a su vez retroalimentan, y se retroalimentan, de todos esos recursos, creando nuevas opciones educativas en diferentes sectores a nivel local, nacional e internacional.

Bibliografía

(11 de Septiembre de 2014). Obtenido de DCB, Matematicas: <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/>

(12 de septiembre de 2014). Obtenido de http://www.w3schools.com/html/html5_intro.asp

(12 de septiembre de 2014). Obtenido de <http://www.w3.org/>

(12 de septiembre de 2014). Obtenido de <https://es.khanacademy.org/>

(13 de septiembre de 2014). Obtenido de <https://www.coursera.org/>

(19 de septiembre de 2014). Obtenido de <http://www.wescheme.org>

(19 de septiembre de 2014). Obtenido de Ambiente de programacion: <http://www.wescheme.org/openEditor>

(19 de septiembre de 2014). Obtenido de Lenguaje: <http://racket-lang.org/>

(19 de septiembre de 2014). Obtenido de API: <http://www.rastersoft.com/OS2/CURSO/APIEXPL.HTM>

(10 de septiembre de 2014). Obtenido de Proyecto Arquimedes: <http://arquimedes.matem.unam.mx/>

(22 de septiembre de 2014). Obtenido de PLT Redex: <http://redex.racket-lang.org/>

(22 de septiembre de 2014). Obtenido de How to Design Programs: <http://www.htdp.org/>

(22 de septiembre de 2014). Obtenido de Realm of Racket: <http://www.realmofracket.com/>

(22 de septiembre de 2014). Obtenido de PLAI: <http://cs.brown.edu/~sk/Publications/Books/ProgLangs/2007-04-26/>

(22 de septiembre de 2014). Obtenido de Picturing Programas: <http://picturingprograms.com/>

(22 de septiembre de 2014). Obtenido de Racket: <http://racket-lang.org/>

(22 de septiembre de 2014). Obtenido de Bootstrap: <http://www.bootstrapworld.org/>

(22 de septiembre de 2014). Obtenido de Realm of Racket: <http://www.realmofracket.com/>

(13 de septiembre de 2014). Obtenido de Bootstrap: <http://www.bootstrapworld.org/>

(septiembre de 2014). Obtenido de GitHub: <https://github.com/plt/racket/wiki/Courses-using-Racket>

(27 de septiembre de 2014). Obtenido de Indiana: <http://www.cs.indiana.edu/classes/c211/>

(27 de septiembre de 2014). Obtenido de Northeastern: <http://www.ccs.neu.edu/course/cs2500f14/>

(27 de septiembre de 2014). Obtenido de Tufts: <http://www.cs.tufts.edu/comp/50PSS-2013f/>

(27 de septiembre de 2014). Obtenido de Vassar: <http://www.cs.vassar.edu/courses/top>

(27 de sept de 2014). Obtenido de CPSC 110: <https://sites.google.com/site/ubccpsc110/>

(27 de septiembre de 2014). Obtenido de CS3613: Programming Languages, Winter: <http://www.cs.unb.ca/~bremner/teaching/cs3613/>

(27 de septiembre de 2014). Obtenido de Computer Science Moodle: <https://cs.berry.edu/moodle/>

(27 de septiembre de 2014). Obtenido de ProgramDesign: <https://www.coursera.org/course/programdesign>

(27 de septiembre de 2014). Obtenido de CS 115: <https://www.student.cs.uwaterloo.ca/~cs115/>

(27 de septiembre de 2014). Obtenido de CS 1410-20: Object-Oriented Programming: <http://www.eng.utah.edu/~cs1410-20/>

(13 de septiembre de 2014). Obtenido de Standards for Mathematics: <http://www.corestandards.org/Math/>

(13 de septiembre de 2014). Obtenido de Standards for Mathematics: <http://www.corestandards.org/Math/>

(13 de septiembre de 2014). Obtenido de Standards for Mathematical Practice: <http://www.corestandards.org/Math/Practice/>

(29 de septiembre de 2014). Obtenido de Evaluacion: http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/operaciones_algebraicas.html

(29 de septiembre de 2014). Obtenido de Evaluación de Aritmética: <https://es.khanacademy.org/mission/arithmic>

(29 de septiembre de 2014). Obtenido de Evaluaciones de Álgebra I: <https://es.khanacademy.org/mission/algebra>

(30 de septiembre de 2014). Obtenido de Álgebra I: <https://es.khanacademy.org/mission/algebra>

(30 de septiembre de 2014). Obtenido de operaciones algebraicas: Cuestionario de Álgebra, http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/operaciones_algebraicas.html (consultado el 29 de septiembre de 2014)

(30 de septiembre de 2014). Obtenido de curso Kiczales: <https://class.coursera.org/programdesign-002>

(30 de septiembre de 2014). Obtenido de Evaluación de Aritmética: <https://es.khanacademy.org/mission/arithmic>

(30 de septiembre de 2014). Obtenido de Ejercicio en WeScheme: <http://www.wescheme.org/openEditor?publicId=JjZp3nDmUv>

(30 de septiembre de 2014). Obtenido de Receta de diseño: http://www.ccs.neu.edu/home/matthias/HtDP2e/part_one.html#%28part._ch~3ahtdp%29

(10 de septiembre de 2014). Obtenido de Youtube: <https://www.youtube.com/>

(11 de septiembre de 2014). Obtenido de ITESM: http://www.itesm.mx/wps/portal?WCM_GLOBAL_CONTEXT=

(11 de septiembre de 2014). Obtenido de coursera: <https://www.coursera.org/unam>

(11 de septiembre de 2014). Obtenido de UDG: <http://www.udg.mx/es>

Bootstrap. (13 de septiembre de 2014). Obtenido de <http://www.bootstrapworld.org/>

CCH. (11 de Septiembre de 2014). Obtenido de <http://www.cch.unam.mx/>

Coleccion Conocimientos Fundamentales. (11 de Septiembre de 2014). Obtenido de <http://www.conocimientosfundamentales.unam.mx/>

Coordinaciones Académicas. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/>

CREA. (11 de septiembre de 2014). Obtenido de <http://www.crea.udg.mx/crea.jsp>

CREADOR. (septiembre de 2014). Obtenido de <http://www.tecnologias.ciep.cga.udg.mx/creador/index.php>

DCB. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/index.html>

DCB. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/Autoevaluacion/index.html>

DCB. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/lgebra.html>

DCB, Algebra. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/CoordinacionesAcademicas/Matematicas/CapsulasAntecedentes/lgebra.html>

c.unam.mx/CoordinacionesAcademicas/Matematicas/Algebra/
DCB, UNAM. (11 de Septiembre de 2014). Obtenido de <http://dcb.fi-c.unam.mx/>
Estandares. (13 de septiembre de 2014). Obtenido de
<http://www.bootstrapworld.org/materials/Spring2014/Standards.shtml>
F.I, UNAM. (10 de septiembre de 2014). Obtenido de UNAM: <http://www.ingenieria.unam.mx/>
Massive Open Online Courses. (10 de Septiembre de 2014). Obtenido de <http://www.mooc.es/ques-un-mooc/>
Michigan. (13 de septiembre de 2014). Obtenido de <https://www.coursera.org/umich>
Proyecto Arquimides. (11 de Septiembre de 2014). Obtenido de <http://arquimedes.matem.unam.mx/>
Proyecto Descartes. (10 de septiembre de 2014). Obtenido de
http://proyectedescartes.org/miscelanea/tematico_algebra.htm
Recursos Educativos para Todos. (11 de Septiembre de 2014). Obtenido de
<http://suayed.unam.mx/recursos/>
Tokio Coursera. (13 de septiembre de 2014). Obtenido de <https://www.coursera.org/utokyo>
UnADM. (11 de septiembre de 2014). Obtenido de <http://www.unadmexico.mx/>
Universidad Nacional Autonoma de Mexico. (11 de Septiembre de 2014). Obtenido de
<http://www.unam.mx/>
Wikipedia. (12 de septiembre de 2014). Obtenido de
http://en.wikipedia.org/wiki/Wikipedia:Ten_things_you_may_not_know_about_Wikipedia
Wikipedia. (12 de septiembre de 2014). Obtenido de <http://es.wikipedia.org/wiki/Wikipedia:Portada>
YouTube. (12 de septiembre de 2014). Obtenido de <https://www.youtube.com/>