



VOLUMEN

8



Revista De Investigación Estudiantil

ILLUMINATE

INGENIERIA DE SISTEMAS

2016



"SEMILLEROS DE INVESTIGACION
ESTUDIANTIL"

REVISTA DE INVESTIGACION ESTUDIANTIL

ILUMINATE VOL. 8

DIRECCIÓN

Lic. Wilma Peñafiel

EDITOR RESPONSABLE:

Lic. Wilma Peñafiel
wpenafiel@ulasalle.edu.bo

Comité Editorial (ULASALLE):

Lic. Martha Heredia
M.Sc. Marcelo Saavedra
Lic. Javier Heredia

Comité Editorial Externo

Lic. Dindo Valdez (Universidad Mayor de San Andrés)
Bolivia

Comité Internacional

Dr. Marcelo Vera (ISTEC, Nuevo México)
Estados Unidos

Diseño y Diagramación:

Herlan Flores

Impresión: Universidad La Salle

Depósito Legal: 4-3-28-09

ISSN 2415-2323

Dirección: Calle Jorge Carrasco, esquina Las Palmas, No 450

Teléfonos: 2723588, 2723598

e-mail: wpenafiel@ulasalle.edu.bo

Noviembre de 2016

La Paz- Bolivia

Presentación

Han transcurrido ocho años, desde la primera publicación estudiantil de la revista Illuminate. Ahora en su octava versión, estamos seguros que seguirá motivado a los estudiantes de la carrera de Ing. de Sistemas a seguir publicando sus trabajos de investigación.

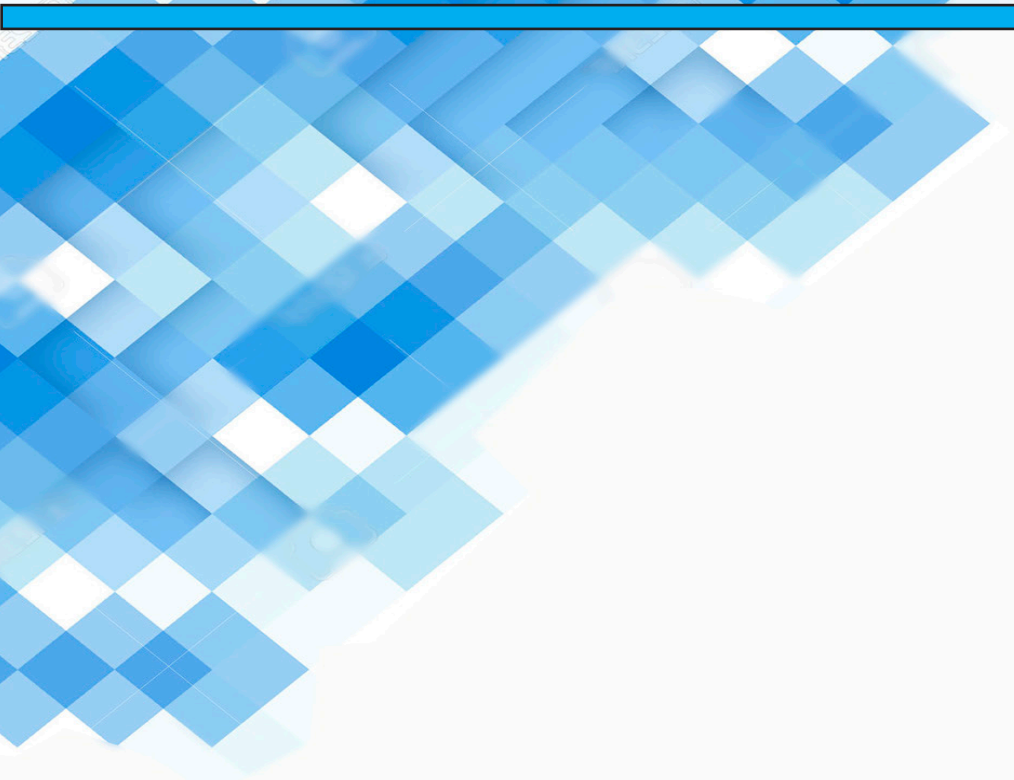
A lo largo de estos ocho años, hemos publicado más de 40 artículos de Investigación de nuestros estudiantes, que al principio tímidamente realizaban sus primeros pasos, y con el tiempo se fueron afianzando en su nueva competencia.

El que hacer investigativo, es complejo y al mismo tiempo es gratificante, porque uno tiene la satisfacción de que alguien leerá su trabajo y de esa forma se sentirá reconocido por el esfuerzo dedicado a su investigación. Los trabajos presentados en esta revista a iniciativa de los estudiantes, tratan sobre aspectos relacionados al entorno de la computación y su vinculación con el medio y la comunidad.

Nuestras felicitaciones a los estudiantes que se atrevieron a dar el primer paso y cuyo trabajo de investigación, lleva nuestro reconocimiento al ser publicados en esta revista.

Queremos agradecer a la Lic. Martha Heredia, Lic. Javier Heredia, Ing. Marcelo Saavedra como comité editorial y a los profesores de la carrera de Ing. de Sistemas, por su orientación en los trabajos de investigación. También un agradecimiento a la Carrera de Ing. De Sistemas de la Universidad La Salle, por promover este tipo de actividades para fortalecer la investigación en nuestra Universidad.

Lic. Wilma Peñafiel
Editora de la Revista Illuminate



CARACTERISTICAS DE LA REVISTA

El Volumen 8, de la Revista Illuminate contiene cinco artículos: de los cuales, Dos artículos son originales, y tres son de revisión de tema.

La revista es del área de Tecnología y abarcan las áreas de Inteligencia artificial, microprocesadores y microcontroladores y por último la teoría de Lenguajes y Autómatas.

Otro aspecto importante a resaltar en este volumen es que los estudiantes comienzan a realizar sus pasos en la investigación con mucha ilusión. El incentivo y la posibilidad que les brinda la revista, no sólo es una motivación para ellos, sino un impulso para continuar en el reto de la investigación. Nuestro reto es lograr que estudiantes de otras Universidades puedan hacernos llegar su trabajos para poderlos publicar y difundir.

Esperamos a través de la revista contribuir al enriquecimiento del conocimiento y de esta manera, favorecer la investigación en nuestro país.

Lic. Martha Heredia
Comité Editorial



Índice

ARTICULOS ORIGINALES

Sistema Experto de Gastronomía Vegetariana	1
Los Microprocesadores, Evolución y Futuro	19

REPORTES DE CASO

Máquinas de Turing y sus aplicaciones	31
Máquinas de Estado Finito	41
Máquinas de Turing en el Arte Musical	59

Sistema Experto de Gastronomía Vegetariana Vegetarian Gastronomy Expert System

Héctor Matías Arhuata
matias_hh@hotmail.com
Jesús Ángel Huayhua Monasterios
jacktenreck@hotmail.com

Carrera de Ingeniería de Sistemas -Universidad La Salle

Resumen

El ambiente y la vida cotidiana de las personas ha estado rodeado de esfuerzo, dedicación y tiempo al trabajo, eso requiere de un desgaste físico para lo cual el ser humano necesita alimentarse, no obstante el diario vivir muestra un panorama distinto de alimentación, el cual demuestra que las comidas chatarras, frituras, grasas realizan un alto índice de obesidad, por tener mayores comodidades y facilidades de la obtención de comida, la mayoría de los habitantes realizan sus preparaciones con recetas.

El sistema de gastronomía vegetariana (SGV), es un sistema experto para la selección de vegetales y control nutricional desarrollado en MATLAB.

El funcionamiento del sistema experto permitirá realizar varias conclusiones de comidas naturales que se dividen en cinco diferentes preparaciones que son: Platos Tradicionales, Sopas o Cremas, Postres, Masas o Panes y jugos el cual reparte resultados dependiendo a las variables que se determinen elegir de una alimentación vegetariana, según la selección que se escoja realizara la búsqueda y tendrá el resultado del objetivo deseado de la alimentación sana

Palabras Clave—Alimentación sana, Base de hechos, Metodología Buchanan, Sistemas expertos

Abstract

The environment and daily life of people has been surrounded by effort, dedication and time to work, that requires a physical wear and tear for which the human being needs to feed, however daily life shows a different view of food, which Shows that junk foods, fried foods, fats perform a high rate of obesity, because they have more amenities and facilities to obtain food, most of the inhabitants make their preparations with recipes.

The vegetarian gastronomy system (SGV) is an expert system for plant selection and nutritional control developed in MATLAB.

The operation of the expert system will allow to make several conclusions of natural foods that are divided into five different preparations that are: Traditional Dishes, Soups or Creams, Desserts, Doughs or Breads and juices which distributes results depending on the variables that are determined to choose from one Vegetarian food, according to the selection that will be made the search and will have the result of the desired goal of healthy eating

Key Words

Healthy Eating, Basis of Facts, Buchanan Methodology, Expert Systems.

Introducción

Los Sistemas expertos [04], (1995) se caracterizan no solo por su arquitectura sino también por su capacidad. La capacidad es la habilidad de emular ciertas funciones inteligentes del ser humano a través de Sistemas Basados en Conocimiento. Estos sistemas son programas de ordenador capaces de representar conocimientos sobre el dominio específico y poder razonar como lo haría el ser humano.



Figura 1: Estructura de un Sistema Experto
Fuente: DURKIN JHON, (1994)

Metodología

El entorno del sistema SGV está basado en la metodología Buchanan, según [05]

Es una estructura de sistemas inteligentes cual tiene procesos que identifican por las siguientes fases:

- Identificación.
Fase que se reconocen aspectos importantes del problema, las características del problema, los recursos disponibles y las metas a alcanzar
- Conceptualización.
Fase que trata de organizar el conocimiento según un esquema conceptual, al mismo tiempo que intentan determinar cómo es el flujo de información durante el proceso de resolución de problemas.
- Formalización.
Fase que trata en traducir los conceptos clave, los sub problemas, y las características del flujo de información, identificados durante la fase anterior, en representaciones formales basadas en herramientas o esquemas.
- Implementación.
Fase que trata de formular reglas, y estructuras de control, que representan los

conceptos y el conocimiento formalizado. El resultado es un programa prototipo que nos permite comprobar si hemos conceptualizado y formalizado bien el conocimiento que el experto tiene sobre el problema.

- Prueba.

Fase que trata de la evaluación del rendimiento del prototipo construido para encontrar errores o anomalías en la base de conocimientos o en los mecanismos de inferencia.

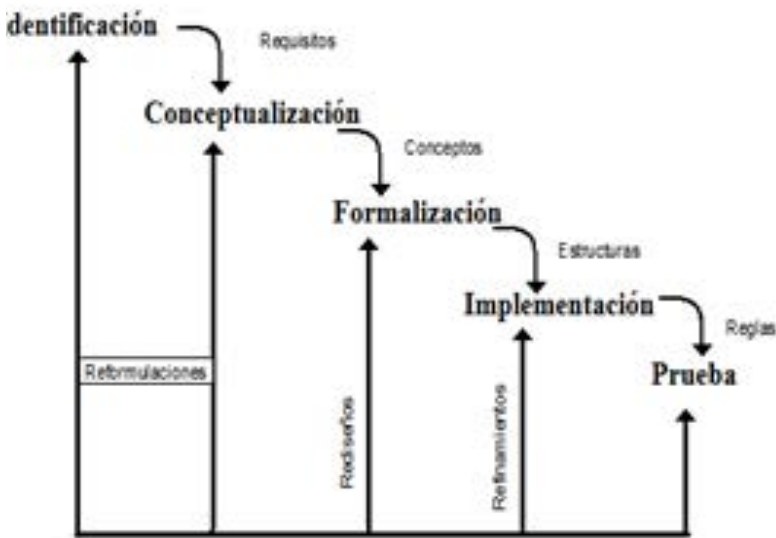


Figura 2. Metodología Buchanan.
Fuente: DURKIN Jhon, 1994

Referentes Conceptuales

1. Sistema Experto

Según [03]: “Se puede entender como una rama de la inteligencia artificial. Estos sistemas imitan las actividades de un humano para resolver problemas de distinta

índole (no necesariamente tiene que ser de inteligencia artificial)” (1991). También se dice que un se basa en el conocimiento declarativo (hechos sobre objetos, situaciones) y el conocimiento de control (información sobre el seguimiento de una acción).

Para que un sistema experto sea herramienta efectiva, los usuarios deben interactuar de una forma fácil, reuniendo dos capacidades para poder cumplirlo: Explicar sus razonamientos o base del conocimiento: los sistemas expertos se deben realizar siguiendo ciertas reglas o pasos comprensibles de manera que se pueda generar la explicación para cada una de estas reglas, que a la vez se basan en hechos.

Estructura básica de un SE

- Un Sistema Experto está conformado por:
- BASE DE CONOCIMIENTOS (BC): Contiene conocimiento modelado extraído del diálogo con el experto.
 - BASE DE HECHOS: (Memoria de trabajo): contiene los hechos sobre un problema que se ha descubierto durante el análisis.
 - MOTOR DE INFERENCIA: Modela el proceso de razonamiento humano.
 - MÓDULOS DE JUSTIFICACIÓN: Explica el razonamiento utilizado por el sistema para llegar a una determinada conclusión.
 - INTERFAZ DE USUARIO: es la interacción entre el Sistema Experto y el usuario, y se realiza mediante el lenguaje natural.

Tipos de Sistema Experto
Principalmente existen tres tipos de sistemas expertos:

- BASADOS EN REGLAS
- BASADOS EN CASOS CBR (Case Based Reasoning).
- BASADOS EN REDES :

En cada uno de ellos, la solución a un problema planteado se obtiene:

- Aplicando reglas heurísticas apoyadas generalmente en lógica difusa para su evaluación y aplicación.
- Aplicando el razonamiento basado en casos, donde la solución a un problema similar planteado con anterioridad se adapta al nuevo problema.
- Aplicando redes bayesianas, basadas en estadística y el teorema de Bayes.

Podemos decir que un Sistema Experto es una Base de Conocimientos (BC), una Base de Hechos (BH) y un Motor (o Máquina) de Inferencias (MI). Por otra parte estos sistemas no se miden por la cantidad de instrucciones o programas sino por la cantidad de reglas que hay contenida en su Base de Conocimientos.

Formas de representación de los conocimientos:

- Reglas de producción
- Redes semánticas
- Marcos (Frames).

La forma de representación más usada es por reglas de producción, también llamadas reglas de inferencias. Casi todos los sistemas expertos están basados en este tipo de representación, ahora nos ocuparemos de los sistemas basados en reglas.

Las reglas de producción son del tipo:

SI Premisa ENTONCES Conclusion (SI A ENTONCES B).
Donde tanto las premisas como la conclusión, no son más que una cadena de hechos conectados por “Y” o por “O”, de forma general sería:

SI Hecho1 Y/O Hecho2 Y/O... HechoN ENTONCES Hecho1 Y/O ... HechoN
Los hechos son afirmaciones que sirven para representar conceptos, datos, objetos, etc. Y el conjunto de hechos que describen el problema es la base de hechos.

Ejemplo de hechos:

Juan es un estudiante
Juan tiene 8 años
el perro es blanco
a María le gusta el cine
Pedro prefiere la película
la edad de Luis es de 25 años
Pedro tiene un salario de 200 pesos

Una regla es una combinación de hechos que permite representar conocimientos y sacar inferencias de los mismos.

Ejemplo de reglas:
R1: SI Juan es estudiante Y Juan tiene 8 años Entonces Juan estudia en la primaria.
R2: SI el perro es blanco Y el perro se llama Dinky ENTONCES el perro es de Juan.
R3: SI a Maria le gusta la película Y Juan prefiere la pelota ENTONCES hacen falta televisores

Observe como partiendo de hechos conocidos que describen algún conocimiento se pueden inferir nuevos hechos (nuevos conocimientos), por otra parte la regla #2 (R2), no tiene por que ser totalmente cierta, existe la posibilidad de que el perro sea de Juan, quizás se puede afirmar, si fuéramos a cuantificar esa posibilidad, que el perro pertenece a Juan con una certeza de un 80%, y por último la regla #3 (R3) es dependiente del contexto, ya que aquí se supone que ambos viven juntos y que los programas de TV coinciden.

La Base de Conocimientos (BC).

Son el conjunto de reglas que permiten representar los conocimientos del dominio de experto donde cada regla aisladamente tiene significado propio. Normalmente los conocimientos son de tipo declarativo por lo cual la BC casi siempre es una descripción de los conocimientos del experto, por lo tanto requiere de algún mecanismo que obtenga las inferencias adecuadas para resolver el problema, alguien que seleccione las reglas y las vaya ejecutando, ese alguien es el motor de inferencias.

El Motor de Inferencias (MI) es un programa de control cuya función es seleccionar las reglas posibles a satisfacer el problema, para ello se vale de ciertas estrategias de control sistemáticas o de estrategias heurísticas.

- Estrategias de control sistemático:
- Encadenamiento hacia adelante o hacia atrás.
- Búsqueda en profundidad o a lo ancho.
- Régimen de control irrevocable o por tentativa.

Estas estrategias son de forma sistemática las cuales deben llevar a la solución del problema. Podemos decir que el control sistemático es un programa de control hecho de forma “algorítmica” que aplican una heurística de propósito general cuya función es una exploración exhaustiva y metódica de la base de conocimientos. Estos mecanismos de control son muy generales y a veces resultan ineficientes ya que siguen una secuencia de búsqueda demasiado rígida, para resolver esto se idearon las estrategias de control heurísticas.

Las estrategias de control heurísticas son programas de control que utilizan una heurística más específica y su función es una selección más restringida orientada por las necesidades del problema. Estas estrategias actúan sobre el control sistemático y en ciertos casos toma el control y dirige la búsqueda hacia ciertos criterios rompiendo así el control sistemático, una vez logrado su objetivo le devuelve el control nuevamente al sistemático. Estrategias de control heurísticas:

- Orden de las reglas.
- Mayor credibilidad en las reglas.
- Menor número de cláusulas no instanciadas.
- Mayor número de conclusiones en las reglas.

Sistema de Alimentación SGV:

El sistema de alimentación SGV es un sistema experto para la alimentación sana y el cuidado nutricional, que ofrece funciones inteligentes. Este sistema selecciona ingredientes para una receta sana, se describen las decisiones de diseño tomadas durante su desarrollo, obteniendo la evaluación de su funcionalidad y usabilidad.

Al referirnos de sistemas expertos, hablamos de sistemas poco escalables, esto por las plataformas que existen para crearlos. En este desarrollo de programación en Matlab provee los diseños, dimensiones, gráficos, estadísticas y muestreos sistemas expertos también la realización de la lógica difusa

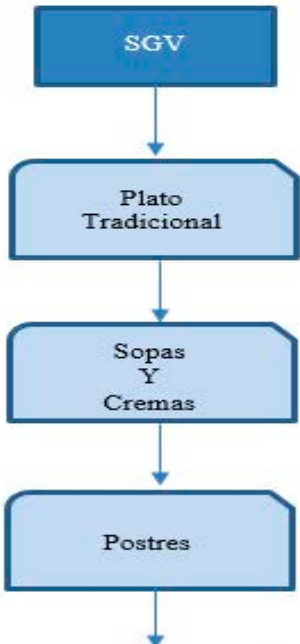
Con esta breve explicación pasamos a la implantación y/o ejecución del SISTEMA EXPERTO DE PROYECTO DE GASTRONOMIA VEGETARIANA utilizando el método Buchanan [05], Pág. 1

2. Estructura del Sistema Experto

2.1. Identificación

El problema es la falta de conocimiento en cuanto a alimentación vegetariana se refiere.

Y los beneficios que se pueden llegar a conseguir por consumir alimentos naturales (Frutas, Cereales y derivados, Verduras y legumbres).



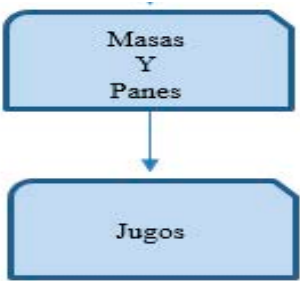
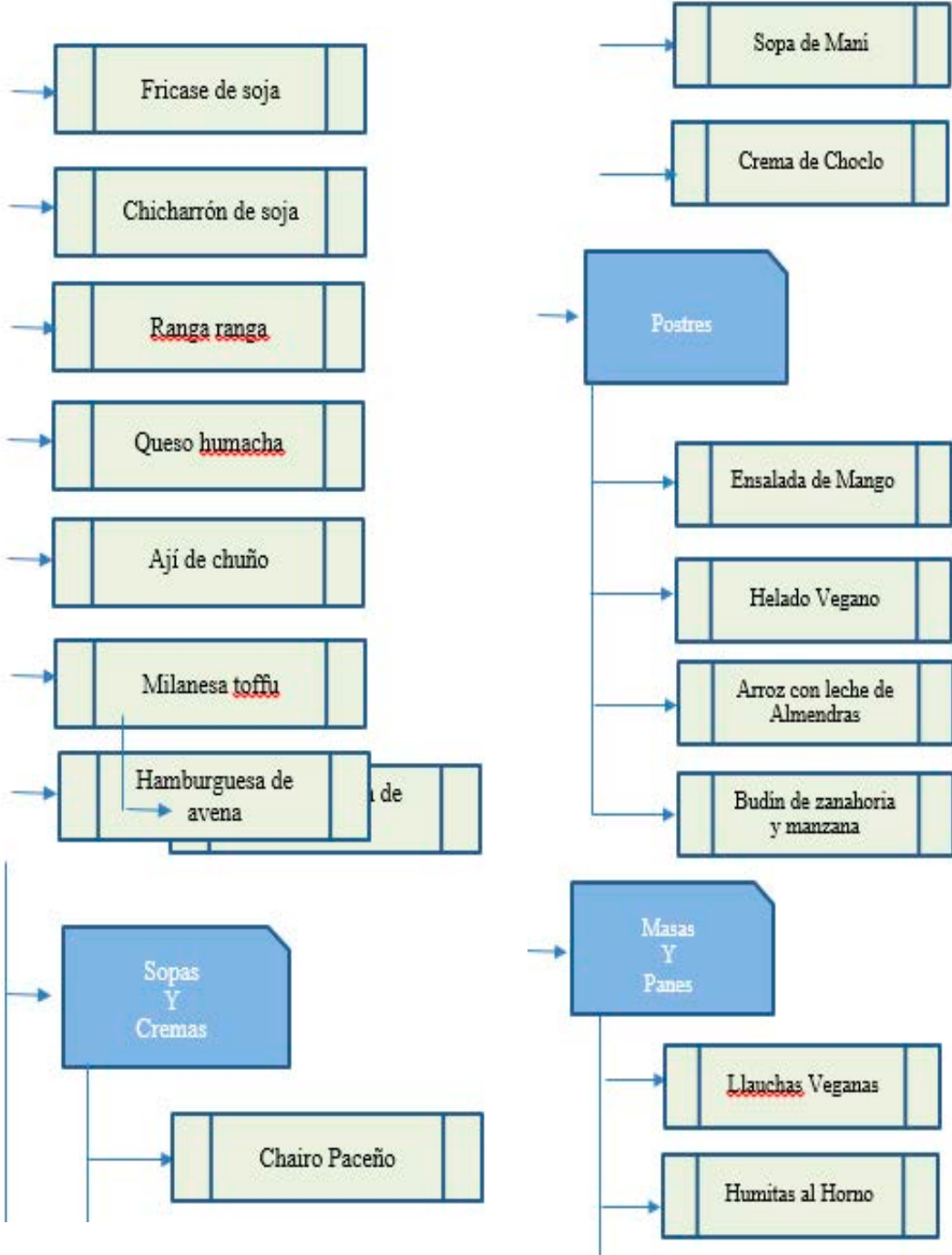
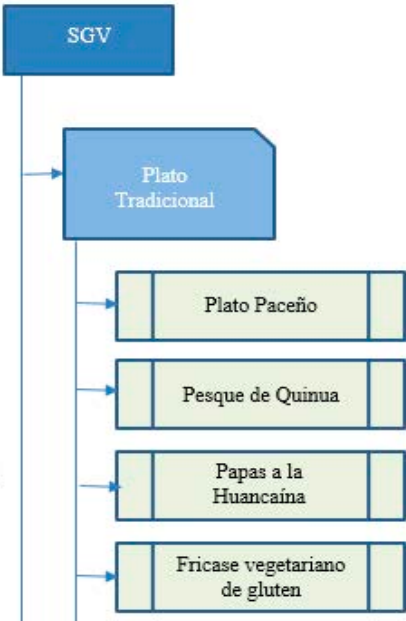


Figura 3. Diagrama Conceptual del Sistema. Fuente propia

Dentro de este punto clasificamos los sistemas que utilizaremos al momento de desarrollar el sistema de gastronomía vegetariana (SGV).
Los dividiremos en 5 grupos ya definidos y de fácil entendimiento para el usuario, de tal forma que indique en que momento sería más conveniente consumir las recetas dadas.

2.2.2. Formalización



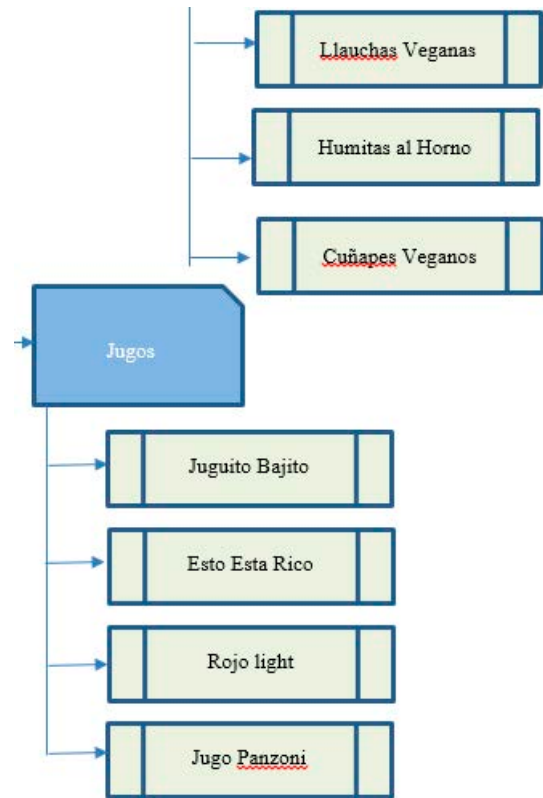


Figura 4. Diagrama Conceptual de recetas. Fuente propia

Añadimos las recetas y clasificamos según al grupo al que pertenezca en el sistema de gastronomía vegetariana (SGV).

2.3. Implementación

Al implantar los datos:

```
% asignamos los checkbox que utilizaremos (por lo general se utiliza cuantos se requiera)
function checkbox1_Callback(hObject, eventdata, handles)
function checkbox2_Callback(hObject, eventdata, handles)
```

```
% se asigna los botones (por lo general se utiliza cuantos se requiera)
function pushbutton2_Callback(hObject, eventdata, handles)

% con ese código limpiamos los checkbox
set(handles.checkbox1,'value',0)
set(handles.checkbox2,'value',0)

% esta función realiza la selección optima de a que receta pertenece
%% jugos
% juguito bajito
if valu36==1 && valu52==1
    if valu33==1 && valu53==1
        if valu38==1
            disp(' juguito bajito ')
        end
    end
end
end

%% masas y panes
% para llauchas veganas
if valu6==1 && valu9==1
    if valu31==1 && valu35==1
        if valu32==1 && valu8==1
            if valu5==1 && valu11==1
                disp(' Llauchas veganas')
            end
        end
    end
end
end
end

%%postres
% ensalada de mango
if valu63==1 && valu64==1
```

```
if valu65==1 && valu22==1
    if valu28==1
        disp('ensalada de mango')
    end
end
end
```

2.4. Prueba

El producto terminado es el siguiente:

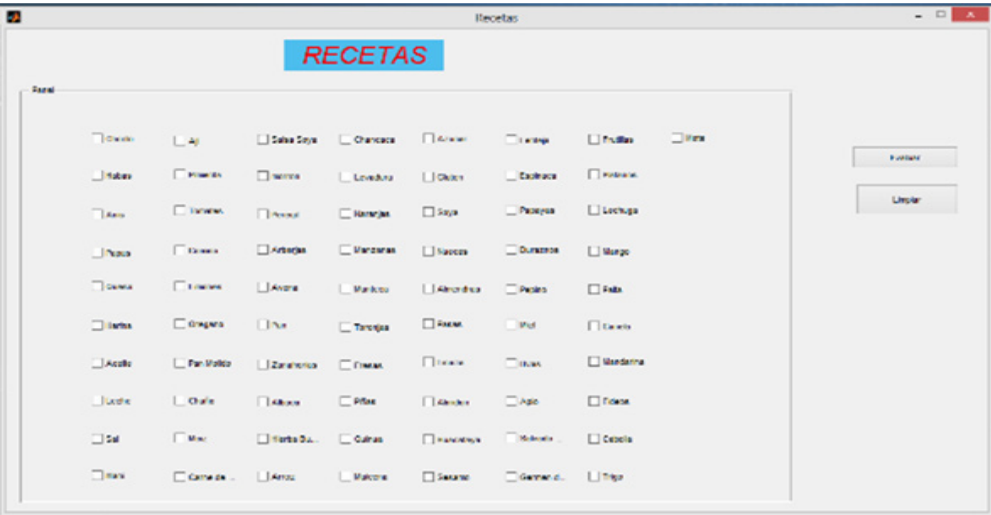


Figura 5. Pantalla de elección de ingredientes del sistema. Fuente propia

3. Resultados y discusión

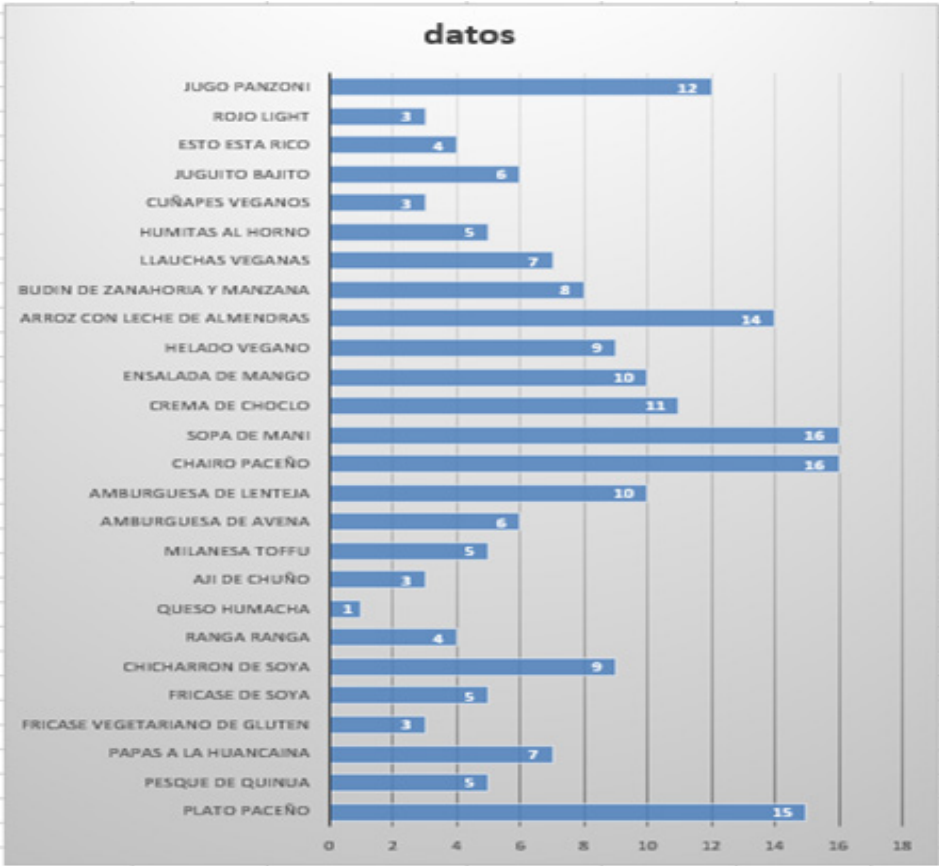


Figura 6. Diagrama de todos las recetas expuestos en el sistema de gastronomía vegetariana (SGV). Fuente: propia

En la figura 6 se puede apreciar un resumen de la elección del menú preferido por cada usuario del sistema, así también la cantidad de veces que se solicitó o seleccionó dicha receta.
Fuente: propia

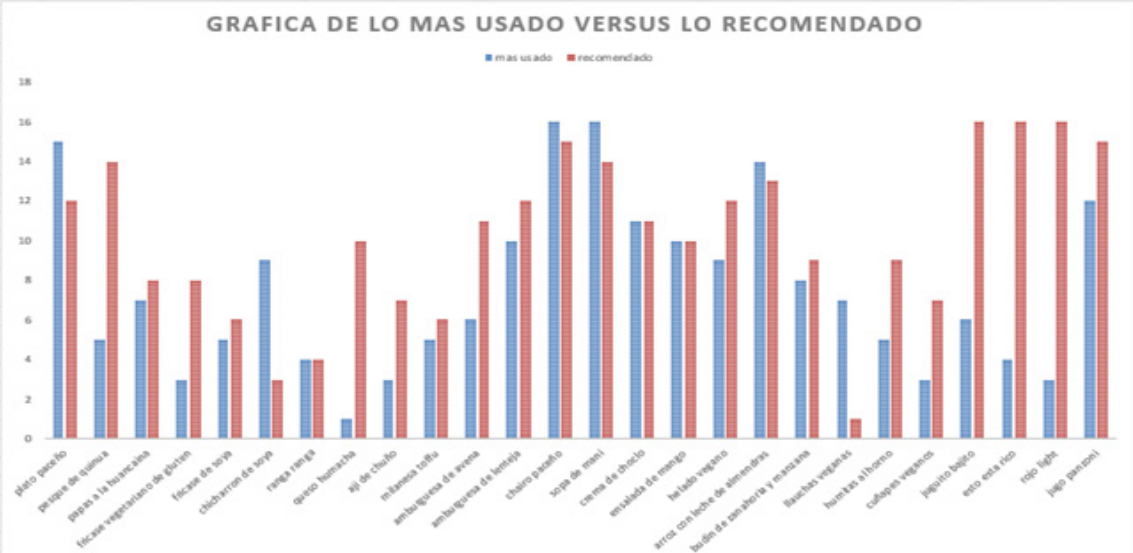


Figura 7. Gráfica de las recetas haciendo una comparación de la frecuencia de lo preferido vs. Lo saludable. Fuente: propia

En la figura 7, la barra azul muestra la frecuencia de elección o preferencia de menú y la barra roja muestra el menú recomendado que es saludable por los nutrientes que ofrece.

Se observa de acuerdo al experimento, que a medida que el sistema va aprendiendo, la recetas saludables se van incrementando.

4. Conclusión

Para concluir el objetivo de este proyecto era crear un sistema que sea capaz de ofrecer una selección de preferencias de menú, ya sea un plato tradicional boliviana, sopas, platos fuertes y postres de acuerdo a la cantidad de ingredientes con que el usuario tenga disponible, incluyendo la sana alimentación.

El sistema fue capaz de ofrecer una receta vegetariana con los pocos ingredientes que se contaba para una buena alimentación.

El problema propuesto se llevó a cabo con la idea de un libro que se encontró y que trataba acerca de la gastronomía vegetariana y se propuso sistematizarla obteniendo buenos resultados.

Los resultados fueron satisfactorios y exitosos por que se llegó a cumplir con la meta propuesta, dado que el sistema ayuda a tomar decisiones a la hora de escoger una receta a preparar y que este se saludable para el usuario.

Cabe recalcar también que este proyecto se podría mejorar y lograr que sea un sistema robusto e incluso poder agregar más recetas.

E incluso poder poner recetas que no sean vegetarianas pero que si ayuden a la buena alimentación de la persona.

Referencias

[01] DURKIN, J. "EXPERT SYSTEMS: DESIGN AND DEVELOPMENT". New York. Maxwell Macmillan. 1994
[02] COHEN, D. "SISTEMAS DE INFORMACIÓN PARA LA TOMA DE DECISIONES". McGrawHill.
[03] SÁNCHEZ, J. "SISTEMAS EXPERTOS: UNA METOLOGIA DE PROGRAMACION". Prentice Hall. México. 1991
[04] BADARO Sebastián, IBÁÑEZ Leonardo Javier y AGÜERO Martin Jorge: "Sistemas Expertos: Fundamentos, Metodologías y Aplicaciones" Pág. 1
[05] DURKIN Jhon. Ingeniería del Conocimiento, Mc Graw Hill, 1995, Pág. 1

Sitios web

[INTER 01] www.uc3m.es
[INTER 02] www.monografias.com
[INTER 03] www.uakron.edu
[INTER 04] www.pucp.edu.pe

Agradecimientos:
Ing. Marcelo Saavedra

Los Microprocesadores, Evolución y Futuro Microprocessors, Evolution and Future

Manuel Augusto Balderrama Alcala, Joel Braulio Gonzales Villca, Christian Cruz Benitez, Kevin Andrés Peñafiel Lucuy, Fernando Mamani, Orlando Zambrana Ramirez, Sendy Quitihuari Arahona

maba_reflex@hotmail.com, br91y677@hotmail.com, cruzitocris2l@gmail.com, kevin_penafiel@hotmail.com, xxxfernandosss@gmail.com, ozambranaramires@gmail.com, sindy.1507.75@gmail.com

Carrera de Ingeniería de Sistemas -Universidad La Salle

Resumen

En el siguiente trabajo se estudió acerca de los microprocesadores, su capacidad de memoria y su máximo alcance de velocidad. Se realizó una recolección de datos, información acerca de los diferentes tipos de microprocesadores ya que al pasar los años se pudieron mejorar y crear más de estos componentes con diferentes capacidades.

Esto significa que para cada componente en su gestión de uso, se le fue mejorando al pasar el tiempo, por lo cual nos concentraremos en los años futuros.

En conclusión obtendremos datos estadísticos los cuales van a determinar con ayuda del modelo de suavizamiento exponencial, la magnitud de mejora de los componentes de microprocesadores.

Palabras claves.

Microprocesadores, memoria, velocidad, suavizamiento exponencial, ley de Moore.

Abstract.

In the next work we study about the microprocessors, their capacity of memory and their maximum reach of speed. A collection of data, information about the different types of microprocessors was made since over the years they could improve and create more of these components with different capacities. This means that for each component in its use management, it has improved over time, so we will focus on future years. In conclusion we will obtain statistical data which will determine with the aid of the exponential smoothing model the magnitude of improvement of the microprocessor components.

Keywords.

Microprocessors, memory, speed, exponential smoothing, Moore's law.

Introducción.

La automatización de las tareas es un hecho, el mundo en que vivimos cambio su accionar, y gran parte de ello se debe a las computadoras, la evolución de la sociedad ahora está marcada por la evolución de las computadoras.

La evolución de las computadoras en sus inicios era regido por la 'la ley de Moore' que decía: "El número de transistores de un microprocesador se duplicará cada dos años", esto indicaba cada cuanto tiempo debería salir un nuevo computador. Esta ley se cumplió según su creador (Cada 50 años), donde los microprocesadores incrementaban su capacidad y reducían su tamaño, pero todo tiene un límite, por influencias económicas, físicas y de investigación las computadoras dejaron de cumplir dicha ley. Ahora nos preguntamos que esperamos de las computadoras y si seguimos un análisis estadístico para saber qué características debería tener una computadora en 10 años. (<http://computerhoy.com/>)

Ahora realizaremos una recolección de datos y trataremos de interpolar datos históricos para saber que computadora nos depara el futuro.

Según los diferentes sitios web investigados se tiene:

- 8086/8080:(<http://www.chw.net/>) El Intel 8086 y el Intel 8088 son los primeros microprocesadores de 16 bits diseñados por Intel. Fueron el inicio y los primeros miembros de la arquitectura x86. El trabajo de desarrollo para el 8086 comenzó en la primavera de 1976 y fue lanzando al mercado en el verano de 1978. El 8088 fue lanzado en 1979.

Esta máquina tenía la capacidad de administrar 1 MB de memoria principal y la velocidad de trabajo era 8 MHz. Con un reloj interno

- 80286: (<http://www.municion.org/>)Luego del fracaso comercial del 80186 que nunca salió a la venta Intel desarrollo y comercializo el 80286 que es un microprocesador de 16 bits de la familia x86, que fue lanzado al mercado por Intel el 1 de febrero de 1982. Cuenta con 134.000 transistores. Al igual que su primo contemporáneo, el 80186, puede ejecutar correctamente la mayor parte del software escrito para el Intel 8086 y el Intel 8088. La administración de memoria principal alcanza a 16 MB y las velocidades de esta máquina eran de 25 MHz.

- 80386:(<https://www.alpertron.com.ar/>)El Intel 80386 es un microprocesador CISC con arquitectura x86. Durante su diseño se lo llamó 'P3', debido a que era el prototipo de la tercera generación x86. El i386 fue empleado como la unidad central de proceso de muchos ordenadores personales desde mediados de los años 80 hasta principios de los 90. Esta máquina contaba con un almacenamiento de memoria 32 MB y su velocidad llegaba hasta 40 MHz.

- Intel 80486:(<https://www.ecured.cu/>)Los Intel 80486 (i486, 486) son una familia de microprocesadores de 32 bits con arquitectura x86 diseñados y fabricados por Intel Corporation y también fabricados mediante licencia o ingeniería inversa por otras empresas como IBM, Texas Instruments,AMD, Cyrix y Chips and Technologies con diseños distintos o clonados. El 80486 contaba con un almacenamiento de memoria de 32 MB, su velocidad llegaba hasta los 100MHz.

- Pentium:(<https://www.alpertron.com.ar/>) Intel Pentium es una gama de microprocesadores de quinta generación con arquitectura x86 producidos por Intel Corporation.

El primer Pentium se lanzó al mercado el 22 de marzo de 1993, con velocidades

iniciales de 60 y 66 MHz, 3.100.000 transistores, cache interno de 8 Kb para datos y 8 KiB para instrucciones; sucediendo al procesador Intel 80486. Intel no lo llamó 586 debido a que no es posible registrar una marca compuesta solamente de números. Esta máquina contaba con un máximo de memoria de almacenamiento de 64 MB y con una velocidad de hasta 300 MHz.

- El Intel Pentium II:(<http://www.pchardware.org/>) Introducido en el mercado el 7 de mayo de 1997. Está basado en una versión modificada del núcleo P6, usado por primera vez en el Intel Pentium Pro. La velocidad de bus era originalmente de 66 MHz, pero en las versiones a partir de los 333 MHz se aumentó a 100 MHz. Poseía 32 KB de memoria caché de primer nivel repartida en 16 KB para datos y otros 16 KB para instrucciones. La caché de segundo nivel era de 512 KB y trabajaba a la mitad de la frecuencia del procesador. El Pentium II integra 7,5 millones de transistores. El voltaje que manejaba era de 2.5 / 2.0. Su memoria de almacenamiento con la cual contaba llegaba hasta los 64Mb.

- Pentium III:(<http://www.pchardware.org/>) El Pentium III es un microprocesador de arquitectura i686 fabricado y distribuido por Intel; el cual es una modificación del Pentium Pro. Fue lanzado el 26 de febrero de 1999. Existen versiones del procesador Intel Pentium III con bus de sistema de 100 MHz ó 133 MHz donde el procesador está disponible a la misma frecuencia específica del núcleo aparte de los 100 MHz y 133 MHz de las versiones del Front Side Bus. El uso de la tecnología de Intel de 0.18-micrones disminuye el tamaño del procesador para un desempeño y movilidad sin precedentes ya que antes era solo hasta 0.25 micrones y que permite incorporar 28 millones de transistores en solo 105 mm cuadrados. Pentium III llegó a tener un máximo de memoria de 128 Mb y además contaba con un máximo de velocidad de hasta 400 MHz.

- Pentium IV:(<http://www.intel.la/>) El Pentium 4 original, denominado Willamette, trabajaba con 128 MB de memoria; y fue lanzado el 20 de noviembre de 2000. Cuenta con 42 millones de transistores Bus de 400 MHz. Soporta frecuencias de bus de 533 MHz y de 400 MHz (esta última para el procesador con HT) lo que permite una velocidad de hasta 1.3GB de datos por segundo entrando/

saliendo del procesador. Bus de direcciones de 36 bits. Bus de datos de 64 bits, con tecnología NetBurst que trae los datos en ráfagas de 4 bloques, lo que implica una lectura de 256 bits de datos en cada ciclo de acceso a memoria externa.

- Micro core:(<http://ark.intel.com/>) Intel Core es un microprocesador de sexta generación lanzado en enero del 2006 por Intel, posterior al Pentium D y antecesor al Core 2 Duo. Dispone de dos núcleos de ejecución lo cual hace de este procesador especial para las aplicaciones de subprocesos múltiples y para multitarea. Puede ejecutar varias aplicaciones exigentes simultáneamente con una velocidad de hasta 1.6 GHz, para juegos con gráficos potentes o programas que requieran muchos cálculos, al mismo tiempo que permite descargar música o analizar el PC con un antivirus en segundo plano. Cuenta con una memoria máxima de hasta 2 GB.

- Micro core duo:(<http://ark.intel.com/>) La marca Intel Core 2 se refiere a una gama de CPU comerciales de Intel de 64 bits de doble núcleo y CPU 2x2 MCM (Módulo Multi-Chip) de cuatro núcleos con el conjunto de instrucciones x86-64, basado en el Core microarchitecture de Intel, derivado del procesador portátil de doble núcleo de 32 bits Yonah. El CPU 2x2 MCM de cuatro núcleos tenía dos matrices separadas de dos núcleos (CPU) -uno junto al otro- en un paquete MCM de cuatro núcleos. El Core 2 relegó la marca Pentium a un mercado de gama medio-bajo, y reunificó las líneas de sobremesa y portátiles, las cuales previamente habían sido divididas en las marcas Pentium 4, D, y M. La diferencia entre este micro y el micro CUAD core es el almacenamiento, ya que Core duo contaba con un almacenamiento de hasta 4 GB y además contaba con una velocidad de 1.6 GHz.

- Micro quad core:(<https://www.ecured.cu/>) Intel Core 2 Quad es una serie de procesadores de Intel con 4 núcleos, lanzados el 2 de noviembre de 2006, asegurando ser un 65% más rápido que los Core 2 Duo disponibles en ese entonces, con una velocidad máxima de hasta 2.33 GHz. Para poder crear este procesador se tuvo que incluir 2 núcleos Conroe bajo un mismo empaque y comunicarlos mediante el Bus del Sistema, para así totalizar 4 núcleos reales, a diferencia del AMD Penos X4 que es un pro.

- Intel core i7 primera generación:(<http://ark.intel.com/>) Intel Core i7 es una familia de procesadores 4 núcleos de la arquitectura Intel x86-64, Los Core i7 son los primeros procesadores que usan la micro arquitectura Nehalem de Intel. Memoria de 3 canales (ancho de datos de 192 bits); cada canal puede soportar una o dos memorias DIMM DDR3.

- Intel core i7-segunda generación: (<http://ark.intel.com/>) Intel lanzo sus procesadores con el nombre clave de Sandy Bridge. Estos procesadores que no tiene grandes cambios en arquitectura con respecto a las Nehalem, pero si son más necesarios para hacerlos más eficientes y rápidos que los modelos anteriores. Con nuevas instrucciones de 256 bits, duplicando el rendimiento y todo lo relacionado con la operación en multimedia.

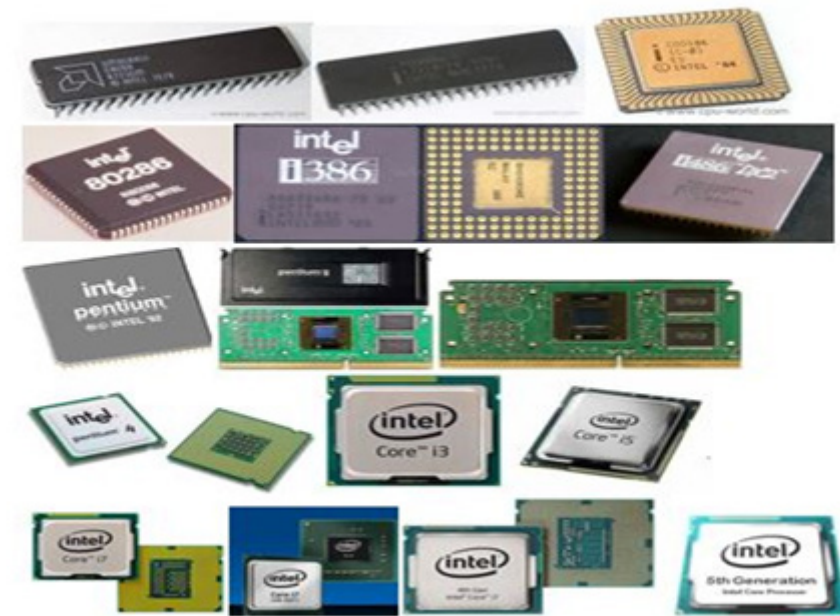
- Intel core i7-tercera generación:(<http://ark.intel.com/>) Ivy Bridge es el nombre clave para los procesadores conocidos como los de Intel Core de tercera generación. Pasamos de los 32 nanómetros de ancho de la Sandy Bridge a los 22 de Ivy Bridge. Esto le permite meter el doble de ellos en la misma área. Un mayor número de transistores significa que puedes poner más bloques funcionales dentro del chip. Es decir capaz de hacer un mayor número de tareas al mismo tiempo.

- Core i7 cuarta generación: (<http://ark.intel.com/>)Es un CPU de 4 núcleos corriendo a 3.40 GHz (overclock 3.9 GHz) 8 Mb de cache, velocidad de bus de 5 GT/s DMI2, de memoria DDR3 o DDR3L con una cantidad máximo de 2 canales de memoria, un máximo de ancho de banda de memoria 25,6 GB/s, procesador gráfico Intel® HD Graphics 4600 que corre a 350 MHz, con una memoria máxima de video de 2GB

- Core i7 quinta generación:(<http://ark.intel.com/>) Un CPU de 4 núcleos corriendo a 3.3 GHz (overclock 3.9 GHz) 8 Mb de cache, velocidad de bus de 5 GT/s DMI2, de memoria DDR3L con una cantidad máximo de 2 canales de memoria, un máximo de ancho de banda de memoria 29,86 GB/s, procesador gráfico Intel® Iris™ Pro Graphics 6200 que corre a 350 MHz, con una memoria máxima de video de 32GB

- Core i7 sexta generación:(<http://ark.intel.com/>) Se trata de una CPU de cuatro núcleos corriendo a 4 GHz (overclock a 4.2. GHz), 8 MB de cache y utilizando el socket LGA 1151 del nuevo chipset Z170. El procesador soporta memorias DDR4 y DDR3L; además, se presenta el nuevo procesador gráfico Intel HD 530 que corre a 350 MHz (llegando hasta 1150 MHz) y con soporte para DirectX 12, con una memoria máxima de video de 64GB.

Imagen 1: Familia de microprocesadores Intel



Fuente: Propia

Objetivo.

Determinar una proyección estadística de las velocidades de funcionamiento del computador como también el tamaño de memoria que maneja está a través de los años.

Fundamento teórico.

Para cumplir con nuestro objetivo de investigación recopilamos información de páginas confiables acerca de los microprocesadores desde su creación hasta la

actualidad.

Según la información recolectada acerca de estos elementos nos concentraremos sobre su velocidad y su capacidad de memoria, más adelante tabulamos la información de tal forma de poder exportar de manera simple al paquete estadístico SPSS.

Basados en la información encontrada para componente de microprocesador según su evolución pronosticada que presentaría en 10 años según el área de estadística usaremos el método de suavización exponencial el cual se basa en el pronóstico formal que se usa más a menudo, por su simplicidad y por la reducida cantidad de datos que requiere, la suavización Para elaborar este pronóstico con suavización exponencial, será suficiente que calculemos un promedio ponderado de los datos más recientes y el pronóstico calculado para dentro del tiempo esperado el cual serán de 10 años.

La presentación de la información realizamos a través de gráficos estadísticos y parámetros estadísticos generados por nuestro modelo.

De aquí podemos deducir los pronósticos deseados.

Resultados.

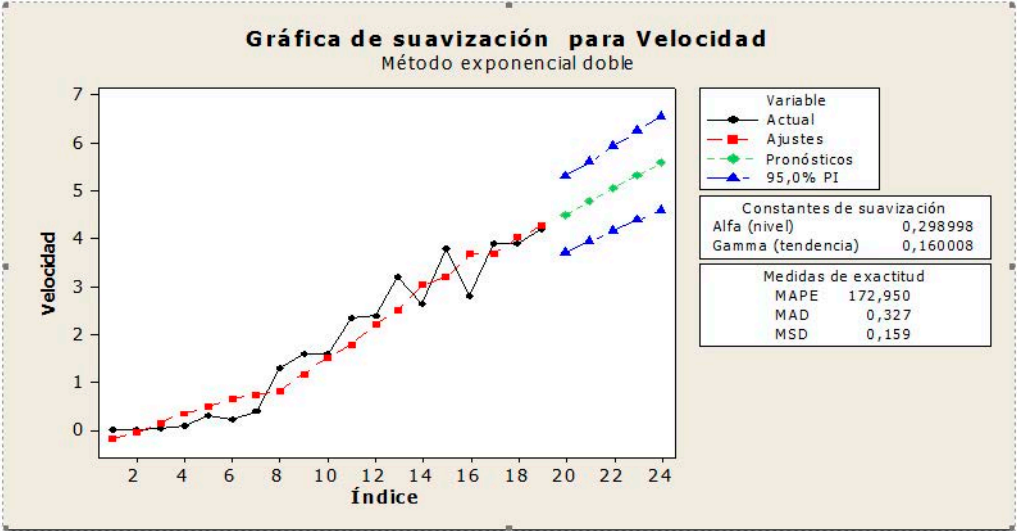
Tabla 1: Información general de microprocesadores

Nro.	Año	Nombre del microprocesador	Máximo de memoria RAM (Gb)	Velocidad GHz
1	1979	8086/8088	0,001	0,008
2	1982	80286	0,016	0,025
3	1985	80386	0,031	0,040
4	1989	80486	0,031	0,100
5	1993	PEMTIUM	0,063	0,300
6	1997	PEMTIUM II	0,063	0,233
7	1999	PEMTIUM III	0,125	0,400
8	2000	PEMTIUM IV	0,125	1,300
9	2006	CORE	2,000	1,600
10	2008	CORE DUO	4,000	1,600
11	2008	CORE QUAD	8,000	2,330
12	2009	i3	8,000	2,400
13	2010	i5	16,000	3,200
14	2010	CORE I7 1GENERACION	16,000	2,660
15	2011	CORE I7 2GENERACION	16,000	3,800
16	2012	CORE I7 3GENERACION	16,000	2,800
17	2013	CORE I7 4GENERACION	16,000	3,900
18	2014	CORE I7 5GENERACION	32,000	3,900
19	2015	CORE I7 6GENERACION	64,000	4,200

PRONOSTICO				
20	2016		107,027	4,513
21	2017		159,647	4,781
22	2018		212,266	5,049
23	2019		264,885	5,318
24	2020		317,505	5,586
25	2021		370,124	5,854
26	2022		422,743	6,122
27	2023		475,363	6,391
28	2024		527,982	6,659
29	2025		580,601	6,927

Fuente propia.

Imagen 2: Imagen de Suavización para la velocidad de funcionamiento del computador



Fuente propia.

Imagen 3: Imagen de Suavización para la capacidad de Memoria de los computadores



Fuente propia

Interpretación.

En los próximos años el crecimiento exponencial de las memorias de almacenamiento y las velocidades de los microprocesadores serán afectados mostrando que la tecnología va mejorando al pasar de los años y que las personas se encuentran con más necesidades con respecto a la tecnología.

De acuerdo a la tabla 1, podemos ver que para 2025, diez años después de haber salido el microprocesador CORE I7 6GENERACION se crearan otros componentes los cuales contarán con más almacenamiento el cual será de 580,601 GB y con una velocidad máxima de 6,927 GHz.

En la imagen 1 y en la imagen 2, según las gráficas se pueden ver los pronósticos obtenidos y muestran según la recta un incremento significativo para posteriores años.

Conclusión.

En conclusión podemos notar que a medida que pasan los años, la tecnología va en incremento mostrando mayores mejoras.

De acuerdo a los datos e información obtenida, según el modelo utilizado para el

pronóstico de los datos de la memoria y de las velocidades podemos afirmar que a través de los años si existe un aumento significativo para la memoria y para las velocidades de los microprocesadores.

A partir del microprocesador CORE I7 6GENERACION los próximos componentes que salgan al mercado tendrán mayores beneficios, contarán con una memoria extensa y una mayor velocidad de funcionamiento para el equipo lo cual será, más adecuado para las personas.

Referencias

- PAÚL DÍAZ, Diego Miguel. Predicción de ventas en el sector retail mediante uso de redes neuronales. Tesis (Ingeniero Civil Industrial). Santiago, Chile: Universidad de los Andes, Facultad de Ingeniería, 2004. 119
- <https://www.alpertron.com.ar/>
- <http://www.pchardware.org/>
- <http://www.usmp.edu.pe/>
- <http://www.todoexpertos.com/>
- <http://www.intel.la/>
- <http://unicrom.com/>
- <http://computerhoy.com/>
- <http://ark.intel.com/>
- <http://www.chw.net/>
- <https://www.ecured.cu/>

Artículo Recibido: 15-09-2016;
Artículo Aceptado: 21-10-2016

Máquinas de Turing y sus aplicaciones Turing Machines and their applications

Cristian Fernando Vilca Gutierrez
crisfer.4217@gmail.com

Carrera de Ingeniería de Sistemas -Universidad La Salle

Resumen

En el presente artículo se sintetizarán conceptos básicos acerca de la Máquina de Turing: componentes de la máquina, funcionamiento y aplicaciones. El trabajo realizado es netamente de investigación y compilación de conceptos de diferentes bibliografías. Se hará énfasis sobre todo en el funcionamiento de la máquina y como esta puede comprenderse mejor con Teoría de Autómatas y Lenguajes Formales, finalmente se mencionarán algunas de sus aplicaciones más importantes en la actualidad.

Palabras claves

Alfabeto, Autómata, Computadora, Formal, Lenguaje, Transición.

Abstract

In this article were synthesized basics about Turing Machine: machine components, operation and applications. The work is purely research and compilation of concepts of different bibliographies. focus mainly on the operation of the machine will be

made and how it can be better understood with Automata Theory and Formal Languages, finally mention some of its most important applications today.

Key words

Alphabet, Automaton, Computer, Formal, Language, Transition.

Introducción

Según Jurado Málaga Elena: La Máquina de Turing (MT) fue introducida por Alan M. Turing en 1936, y puede considerarse como un modelo abstracto que formaliza la idea Intuitiva de algoritmo. Es un modelo matemático para representar a una maquina teórica. A pesar de su simplicidad tiene el mismo poder computacional de una computadora de propósito general. La Máquina de Turing, es interesante, sobre todo, por el conjunto de lenguajes que permite reconocer y también generar (lenguajes recursivamente numerables) y por el conjunto de funciones que puede computar (funciones calculables). (2008, pag. 107)

Su inventor, Alan M. Turing sostiene que la maquina no fue diseñada como una tecnología de computación práctica, sino como un dispositivo hipotético que representa una máquina de computación.

Se caracteriza por ser un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo a ciertas reglas.

Se puede afirmar que las Maquinas de Turing son una buena formalización del concepto de algoritmo por:

Cada programa de una máquina de Turing puede ser implementado.
Todos los algoritmos conocidos han podido ser implementados en una

Maquina de Turing.

Tesis de Crurch: ALGORITMO = MAQUINA DE TURING

En resumen, una Maquina de Turing es un modelo matemático que consiste en un autómata capaz de implementar cualquier problema matemático expresado por medio de un algoritmo.

Referentes conceptuales.

Definición formal

Formalmente una Maquina de Turing puede definirse como una sex-tupla:

$$M = (\Gamma, \beta, Q, \delta, s, h)$$

Donde:

Γ : es el alfabeto de la cinta.

β : es el símbolo en blanco de la cinta

Q : conjunto de estados de la máquina.

δ : es la función de transición.

s : es el estado inicial de la máquina.

h : es el estado final de la máquina.

Según JURADO MÁLAGA Elena : Se dice que una Máquina de Turing (M) acepta una cadena de entrada $w \in \Gamma^*$, si partiendo del estado inicial s y en la cinta se encuentra la cadena w seguida de blancos (β), la Máquina M procesa la cadena y termina en el estado de parada (h).

Se define el lenguaje aceptado por una Máquina de Turing L (M), como el conjunto de todas las cadenas w aceptadas por la máquina.

Los lenguajes aceptados por las Máquinas de Turing se denominan recursivamente enumerables.

El valor computado por una Máquina de Turing para una entrada w se define como el contenido de la cinta cuando la máquina alcanza el estado de parada.

Puesto que pueden existir entradas w para que la Máquina de Turing no alcance nunca un estado final, la función computada por una Máquina de Turing es una función parcial.

Funcionamiento

La Máquina de Turing está formada por una unidad de control que puede describirse mediante un autómata finito, una cinta de lectura y escritura, que tiene un comienzo a la izquierda y se extiende indefinidamente a la derecha, y un cabezal, que indica la posición de la cinta sobre la que trabaja la máquina en cada paso.

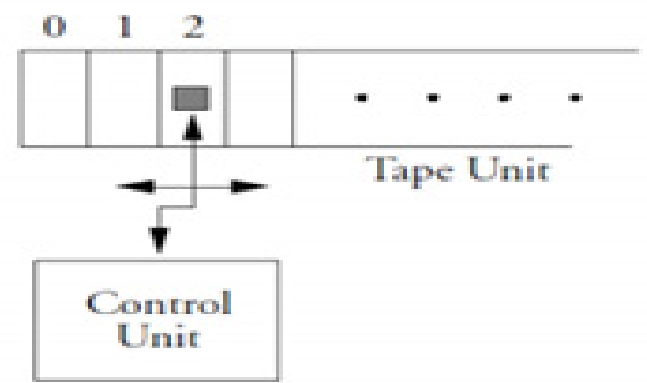


Fig. 1. Máquina de Turing (JURADO MALAGA Elena)

Inicialmente la cinta contiene la cadena de entrada de la máquina (w) seguida de celdas en blanco (β),

En cada paso la maquina lee el contenido de la celda sobre la que se encuentra el cabezal, realiza un cambio de estado de la unidad de control, y escribe un dato sobre la celda o mueve el cabezal de izquierda a derecha. La máquina termina al alcanzar un estado especial llamado estado de parada (halt).

La Máquina de Turing puede considerarse como un autómata capaz de reconocer lenguajes formales. En este sentido es capaz de reconocer los lenguajes recursivamente enumerables. Dada una cadena w , la máquina puede decir si pertenece al lenguaje o no, generando la salida adecuada.

Se dice que el lenguaje es decidable si se puede construir una Máquina de Turing que alcance siempre el estado final y reconozca el lenguaje.

Tipos de Máquina de Turing

Existen varios tipos de Máquinas de Turing, (Jurado Málaga Elena, 2008) todas caracterizadas por tener un comportamiento similar, entre ellas están:

Máquina de Turing con cinta infinita a ambos lados

Esta modificación de denota al igual que una Máquina de Turing sencilla, lo que la hace diferente es que la cinta es infinita tanto por la derecha como por la izquierda.

Máquina de Turing con Cinta Multiplista

Es aquella mediante la cual cada celda de la cinta de una maquina sencilla se divide en subceldas. Cada celda es capaz de contener varios símbolos de la cinta. Se dice que la cinta tiene múltiples pistas porque cada celda de esta Máquina de Turing contiene múltiples caracteres, el contenido de las celdas de la cinta puede ser representado mediante n-tuplas ordenadas. Los movimientos que realice esta máquina dependerán de su estado actual y de la n-tupla que represente el contenido de la celda actual. Debe mencionar que posee un solo cabezal al igual que una Máquina de Turing sencilla.

Máquina de Turing Multicinta

Las Máquinas de Turing con más de una cinta consisten en un control finito con k cabezales lectores/escritores y k cintas. Cada cinta es finita en ambos sentidos. La Máquina de Turing Multicinta define su movimiento dependiendo del símbolo que está leyendo cada uno de sus cabezales, las reglas de sustitución para cada uno de los símbolos y dirección de movimiento para cada uno de los cabezales. Inicialmente empieza con la entrada en la primera cinta y el resto de las cintas en blanco.

Máquinas de Turing Multidimensional

Una Máquina de Turing multidimensional es aquella cuya cinta se extiende infinitamente en más de una dirección, ejemplo más básico sería el de una maquina bidimensional cuya cinta se extendería infinitamente hacia arriba, abajo, izquierda o derecha.

Máquinas de Turing No Deterministas

Es una Máquina de Turing en la que pueden existir varias transiciones a partir del mismo estado y lectura del cabezal. Esto significa que dado un estado y un símbolo de entrada, es posible elegir la transición a efectuar entre varias operaciones. Una Máquina de Turing No Determinista se puede modelar como una Máquina de Turing con una entrada adicional que permite seleccionar la transición a efectuar entre las varias posibilidades a cada paso.

Se pueden reescribir las transiciones añadiendo nuevos estados de manera que en cada caso la elección se efectúe solo entre dos opciones, con lo que el selector

podría ser una señal de un bit.

Todo lenguaje aceptado por una Máquina de Turing No Determinista puede ser aceptado por una Máquina de Turing Determinista.

Máquina de Turing Universal

Es una Máquina de Turing capaz de simular el comportamiento de cualquier Máquina de Turing sobre cualquier cadena de entrada.

Para poder construir una Máquina de Turing Universal es necesario definir una codificación de las Máquinas de Turing, de manera que se pueda controlar a la máquina por medio de su código.

Aplicaciones:

Según Jhon E. Hopcroft et al (2006), la Máquina de Turing puede ser aplicada de diferentes maneras, como por ejemplo:

Teoría de la Computación

Es una rama de las matemáticas y de las ciencias de la computación que centra su interés en las limitaciones y capacidades fundamentales de las computadoras. Específicamente esta teoría busca modelos matemáticos que formalizan el concepto de hacer un cómputo y la clasificación de problemas de acuerdo a su grado de complejidad.

Alan Turing demostró con su Máquina que existen problemas imposibles de ser resueltos algorítmicamente, siendo el Problema de la Parada el más importante. Para estos problemas no existe ni existirá ningún algoritmo que los pueda resolver, no importando la cantidad de tiempo o memoria se disponga en una computadora. Asimismo, con la llegada de las computadoras modernas se constató que algunos problemas resolubles en teoría eran imposibles en la práctica, puesto que dichas soluciones necesitaban cantidades irrealistas de tiempo o memoria para poderse encontrar

Problema De La Parada

El problema de la parada o problema de la detención para máquinas de Turing consiste en: dada una MT M y una palabra w , determinar si M terminará en un

número finito de pasos cuando se ejecuta usando w como entrada.

Alan Turing demostró que este problema es indecidible, ninguna máquina de Turing lo puede resolver.

Máquinas Oráculo

La máquina con oráculo, es una Máquina de Turing equipada con un oráculo que es capaz de contestar preguntas sobre la pertenencia a un conjunto específico de números naturales.

Una máquina oráculo con el "conjunto parada" en su oráculo puede computar la función del problema de la parada. Mientras este sería un ejemplo trivial del uso del conjunto oráculo, muchas otras funciones de interés pueden ser computadas utilizando el oráculo de la función de la parada. De hecho, esto permite que todas las funciones recursivamente enumerables sean computables.

Máquinas de Turing como generadoras de Lenguajes

Si pueden reconocer lenguajes, también son capaces de generarlos

Hipótesis de Church-Turing

"Cualquier computación que pueda ser realizada por medios mecánicos puede ser realizada por una Máquina de Turing."

No es una afirmación que pueda ser probada, sino más bien una definición o explicación del concepto de algoritmo.

Parece ser una definición adecuada:

Otros formalismos propuestos de dicho concepto (funciones recursivas, sistemas de Post, sistemas de reescritura, etc.) no son más expresivos que las Máquinas de Turing.

Nadie ha sugerido ningún problema que pueda ser resuelto por lo que intuitivamente se considera un algoritmo, y que no pueda ser implementado en una Máquina de Turing.

Métodos.

El método que se utilizó para este artículo fue meramente investigativo, se consultaron bibliografías y se realizó una compilación de ambas.

Una Máquina de Turing está compuesta por los siguientes componentes:

Una Cinta, se divide en celdas una al lado de la otra. Cada celda tiene un símbolo de un alfabeto finito. El alfabeto contiene un símbolo especial llamado “blanco” representado por la letra griega $\square$. La cinta es extensible tanto a la derecha como a la izquierda, es decir la máquina suministrará cuanta cinta sea necesaria para su cómputo. Las celdas que no se hayan llenado previamente, se asume que están con el símbolo blanco.

Un Cabezal, encargado de escribir símbolos en la cinta y mover la cinta a la izquierda o a la derecha (siempre una celda a la vez).

Un Registro de Estado, encargada de almacenar el estado de la máquina. Existe un estado inicial en el que cada registro de estado se inicia.

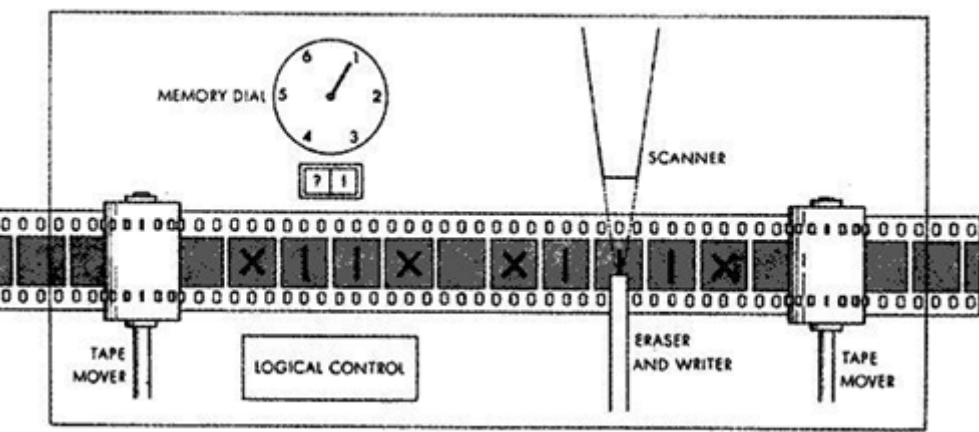


Fig. 2 Lógica de una Máquina de Turing

Resultados y discusión

Una Máquina de Turing puede simular la lógica de cualquier algoritmo de computador y es muy útil para explicar el funcionamiento del mismo. También es posible que una Máquina de Turing simule a otra Máquina de Turing.

(MT Universal)

Es un reconocedor de lenguaje más general que cualquier autómata finito o autómata de pila debido a que puede reconocer lenguajes regular como también lenguajes libres de contexto.

Conclusiones:

Las Máquinas de Turing son muy importantes para comprender de una manera más detallada los Autómatas y los Lenguajes Formales, es muy peculiar comparar una Máquina de Turing con cualquier algoritmo por su similitud. Es importante resaltar el genio del Sr. Turing al poder haber desarrollado una Máquina tan compleja y con tantas aplicaciones prácticas que perduran en la actualidad, con componentes sencillos y una lógica sensacional logro convertirse en uno de los padres de la computación y la informática.

“Un hombre provisto de papel, lápiz y goma, y con sujeción a una disciplina estricta, es en efecto una máquina de Turing universal.”

Alan Turing

Referencias.

ELENA JURADO Malaga. Teoría de Autómatas y Lenguajes Formales
Maquinas de Turing, Modelos Avanzados de Computacion Universidad de Huelva-España
JHON E. HOPCROFT – RAJEEV MOTWANI – JEFFREY D. ULLMAN. Teoría de Autómatas y Lenguajes de Computación
Teoría de Autómatas y Lenguajes Formales. Universidad Rey Juan Carlos

Artículo Recibido: 02-09-2016;
Artículo Aceptado: 24-10-2016

Máquinas de Estado Finito Finite State Machines

Mariluna Alavi, Sofia Aliaga, Monica Murga
marilunaalavichoque@gmail.com, sjaslinkina@gmail.com,
Mon.bramur@gmail.com

Carrera de Ingeniería de Sistemas -Universidad La Salle

Resumen

Conocer el funcionamiento de una máquinas de estados finitos es fundamental ya que es el único tipo de máquinas de estados q se puede modelar a un computador en la actualidad, se mencionara una breve descripción sobre las mismas, su definición formal y las relaciones entre ellas. En otras palabras, observaremos una máquina capaz de seguir una secuencia finita de pasos al introducir un conjunto de datos en ella, solo se puede leer un dato en cada paso que se realice, por tanto el número de pasos a seguir está dado por el número de datos a introducir. Cada entrada diferente genera una salida diferente, pero siempre el mismo resultado con los mismos datos de entrada.

Palabras claves

Estado, Autómata, Síncronas, Asíncronas, Nodos, Ramificación, Diagrama.

Abstract

Knowing the operation of a finite state machines is essential because it is the only type of state machines that can be modeled on a computer today, a brief description of them, the formal definition and the relationships between them are mentioned. In other words, we see a machine capable of following a finite sequence of steps by introducing a set of data in it, you can only read a data in each step is performed, so the number of steps is given by the number entering data. Each different input generates a different output, but always the same result with the same input data.

Key words

State, Automaton, Synchronous, Asynchronous, Nodes, Branch, Diagram.

Introducción

Las máquinas de estado finito son una herramienta muy útil para especificar aspectos relacionados con tiempo real, dominios reactivos o autónomos, computación reactiva, protocolos, circuitos, arquitecturas de software, etc.

El modelo de FSM es un modelo que posee sintaxis y semántica formales y que sirve para representar aspectos dinámicos que no se expresan en otros diagramas.

Una Maquina de estado Finito es una lógica matemática. Es esencialmente un programa de computadora: representa una secuencia de instrucciones a ser ejecutadas, donde cada instrucción depende del estado actual de la máquina y del actual estímulo.

Las posibles entradas al sistema son una secuencia de símbolos seleccionadas desde un conjunto finito I de símbolos de entrada, y las salidas resultantes son secuencias de símbolos escogidas desde un conjunto finito Z de símbolos de salida.

Cualquier “caja negra” que produce un símbolo de salida cuando un símbolo de entrada es aplicado y que satisface las propiedades anteriormente mencionadas es llamada Máquina Secuencial o Transductor Finito.

La selección de un conjunto de estados para representar una máquina dada no es un proceso único, esto no es una limitación muy seria, ya que el principal objetivo es describir el comportamiento general entrada-salida de la máquina en vez de su construcción.

Referentes conceptuales. Una máquina de estados finitos en un modelo abstracto para la manipulación de símbolos, nos permiten saber si una cadena pertenece a un lenguaje o nos pueden generar otro conjunto de símbolos como resultado.

Llamaremos una Máquina de Estados Finitos como Autómata Finito, el hecho es que un Autómata y una Maquina de Estados Finitos son lo mismo, podemos utilizar ambos términos de forma indistinta.

Los Autómatas se caracterizan por tener un Estado inicial, reciben una cadena de símbolos, cambian de estado por cada elemento leído o pueden permanecer en el mismo estado. También tienen un conjunto de Estados Finales o Aceptables que nos indican si una cadena pertenece al lenguaje al final de una lectura.

Los Autómatas se clasifican en 2 tipos:

- Autómata Finito Determinista.
- Autómata Finito no Determinista.

Siempre llamamos un Autómata como Autómata Finito, esto nos puede llevar a pensar que existe algún tipo de Autómata Infinito, lo cual no tiene mucho sentido pensar en un tipo de Maquina que tiene un conjunto infinito de estados, pero aún se discute su utilidad para propósitos prácticos. Un “Autómata Infinito” tiene cintas infinitas o registros de almacenamiento de capacidad ilimitada, esto le da el carácter de infinito.

- Un autómata finito M consta de 5 elementos:
- 1.- un conjunto finito A de símbolos de entrada.
 - 2.- un conjunto finito S de estados internos.
 - 3.- un sub-conjunto T de S (cuyos elementos se llaman estados de aceptación).
 - 4.- un estado inicial q0 en S
 - 5.- una función de estado próximo F de S*A en S.

El autómata M se denota por $M=\langle A,S,T,q_0,F \rangle$ cuando queremos designar sus 5 partes.

Ejemplo: lo siguiente define un autómata finito con dos símbolos de entrada y tres estados.

$A=\{a, b\}$, símbolo de entrada

$S=\{q_0,q_1,q_2\}$, estados

$T=\{q_0,q_1\}$, estados de aceptación q0 estado inicial

La función de estado próximo F: $s \rightarrow a$, s definida por la siguiente tabla:

	a	b
q0	q0	q1
q1	q0	q2
q2	q2	q2

Figura 1: Tabla de función de estado

Podemos describir concisamente un autómata finito M por su diagrama de estado como se hizo con las máquinas de estado finito pero ahora usaremos círculos dobles para estados de aceptación, los estados se denotaran por medio de círculos de los cuales uno será el de inicio y los demás pueden o no ser estados finales, también saldrán de los estados flechas el cual contendrá un símbolo para el próximo estado. El diagrama es el siguiente:

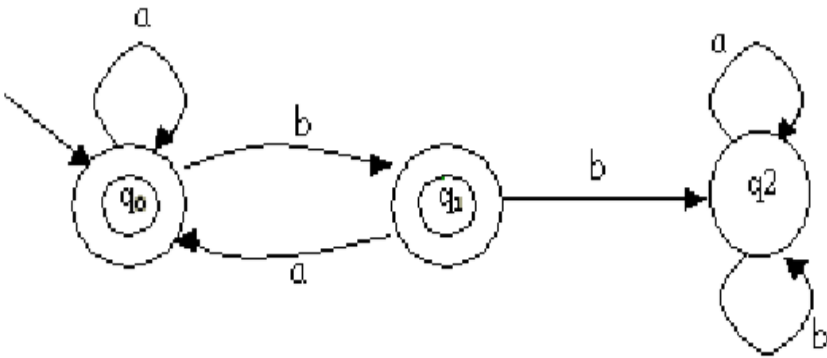


Figura 2: Diagrama autómata finito

Métodos.

Las máquinas de estado pueden ser:

SÍNCRONAS: Necesitan de la intervención de un pulso de reloj. Si la entrada participa también en la salida se denomina Máquina de estado de Mealy, y si no participa se denomina de Moore.

ASÍNCRONAS: No necesitan de la intervención de un pulso de reloj. Estos

circuitos evolucionan cuando cambian las entradas. (Wikipedia, s.f., pág. https://es.wikipedia.org/wiki/M%C3%A1quina_s%C3%ADncrona)

Ahora bien también mencionaremos las diferencias entre:

Un autómata finito o máquina de estado finito es un modelo matemático de un sistema que recibe una cadena constituida por símbolos de un alfabeto y determina si esa cadena pertenece al lenguaje que el autómata reconoce.

Un Autómata finito es similar a una máquina de estado finito sin embargo lo que caracteriza a una de la otra es que los autómatas finitos solo tienen dos estados, un estado interno y uno de rechazo.

Un autómata finito está formado por una quintupla $\langle A, S, T, q_0, F \rangle$ y la máquina de estado finito está formado por una cuádrupla de la forma:

$$\langle Q, \Sigma, A \subseteq Q \times \Sigma \times Q, q_0, \delta \rangle$$

Como diseñar una máquina de estado finito:

Paso 1: Describir la maquina en palabras.- cuando describimos palabras es similar a cuando realizamos el algoritmo de un programa, hay que escribir en detalle todo lo que queremos que realice esta máquina sin obviar comandos o decisiones que tomar.

Paso 2: Dibujar el diagrama.- es una parte importante ya que nos ayudar a entender lo que queremos construir, este debe contener todas las posibles entradas, salidas y valores.

Paso 3: Seleccionar números para representar estados y valores.- es necesario representar los valores en números binarios, después de puede cambiar por otros símbolos y valores a los que representa.

Paso 4: Escribir una tabla de verdad.- esto nos ayudar a saber si el diagrama está bien elaborado.

Paso 5: Dibujar para visualizar el circuito.- al dibujar podremos darnos cuenta si es posible simplificarlo, además que un circuito puede tener sub-circuitos.

(SlideShare, 2010, págs. Diapositiva 8 al 14 <http://es.slideshare.net/junito86/presentacion-2-maquinas-de-estado-finito>)

Resultados y discusión.

La aplicación de los autómatas de estados finitos nos da como resultado las funciones de:

- Adquirir datos del proceso por medio de las entradas digitales y análogas
- Tomar decisiones en base a las reglas de programación
- Almacenar datos en memoria
- Generar ciclos de tiempo
- Realizar cálculos matemáticos
- Actuar sobre dispositivos externos mediante las salidas digitales y análogas.
- Comunicarse con otros sistemas externos.

Las ventajas que podemos mencionar sobre los autómatas son:

- La variedad de modelos existentes
- El ahorro de tiempo en la elaboración de proyectos.
- El bajo costo, tanto como de implementación, como en la modificación de acciones
- El mantenimiento económico por tiempos de paro es reducido.

Los usos de los autómatas en la actualidad son:

- En el control de ascensores.
- En la industria automotriz.
- También pueden realizar operaciones aritméticas.
- Control de operaciones de semáforos.
- Control de nivel de líquidos de un tanque.
- Control de sistemas de envasado.
- Control de una maquina taladora.

Algoritmos de los estados finitos:

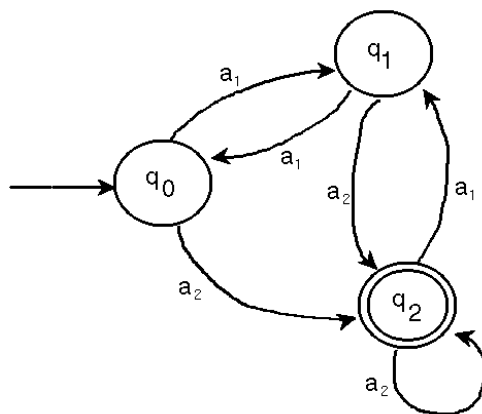


Figura 1: Diagrama de Transición de un Autómata Finita

- Edward Forrest Moore (23 de noviembre de 1925, Baltimore, Maryland-14 de junio de 2003, Madison, Wisconsin) fue un profesor estadounidense de matemáticas y ciencias de la computación, inventor de las máquinas de estados finitos de Moore y uno de los pioneros de la vida artificial.

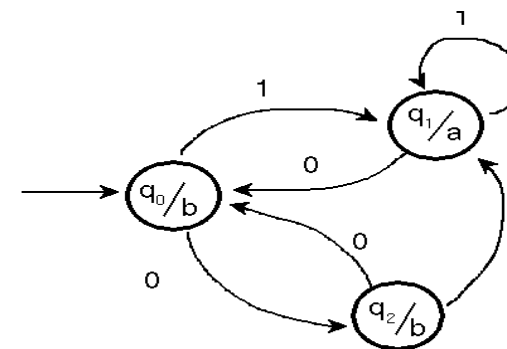


Figura 2: Ejemplo de una Máquina de Moore

Una Máquina de Moore es un autómata de estados finitos para el cual la salida en un momento dado sólo depende de su estado en ese momento, mientras la transición al siguiente estado depende del estado en que se encuentre y de la entrada introducida. El diagrama de estados para una máquina Moore incluirá una señal de salida para cada estado. Comparada con la Máquina de Mealy, la cual mapea transiciones en la máquina a salidas. (Wikipedia.org, 2016)

Definición formal:

Una máquina de Moore puede ser definida como una 6-tupla $\{ S, S_0, \Sigma, \Lambda, T, G \}$ consistente de

- un conjunto finito de estados (S)
- un estado inicio (también llamado estado inicial) S_0 el cual es un elemento de (S)
- un conjunto finito llamado alfabeto entrada (Σ)
- un conjunto finito llamado el alfabeto salida (Λ)
- una función de transición ($T : S \times \Sigma \rightarrow S$) mapeando un estado y una entrada al siguiente estado

- una función salida ($G : S \rightarrow \Lambda$) mapeando cada estado al alfabeto salida. (Wikipedia.org, 2016)

- George H. Mealy (31 de diciembre de 1927, 21 de diciembre de 2010 en Situate, Massachusetts) fue un estadounidense matemático, profesor de la universidad de Harvard e inventor del mismo nombre Autómata de Mealy.

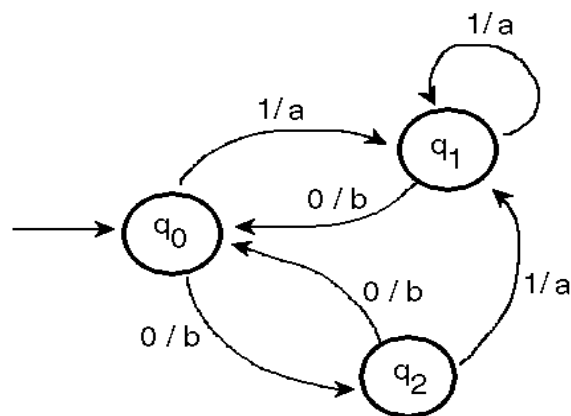


Figura 3: Máquina de Mealy

Una Máquina de Mealy es un tipo de máquina de estados finitos que genera una salida basándose en su estado actual y una entrada. Esto significa que el Diagrama de estados incluirá ambas señales de entrada y salida para cada línea de transición. En contraste, la salida de una máquina de Moore de estados finitos (el otro tipo) depende solo del estado actual de la máquina, dado que las transiciones no tienen entrada asociada. Sin embargo, para cada Máquina de Mealy hay una máquina de Moore equivalente cuyos estados son la unión de los estados de la máquina de Mealy y el Producto cartesiano de los estados de la máquina de Mealy y el alfabeto de entrada. (Wikipedia.org, 2016)

Definición formal:

Una máquina de Mealy es una 6-tupla, $(S, S_0, \Sigma, \Lambda, T, G)$, consistiendo en un

conjunto finito de estados (S) un estado inicial S_0 el cual es un elemento de (S) un conjunto finito llamado el alfabeto entrada (Σ) un conjunto finito llamado el alfabeto salida (Λ) una función de transiciones ($T : S \times \Sigma \rightarrow S$) una función de salida ($G : S \times \Sigma \rightarrow \Lambda$). (Wikipedia.org, 2016)

- Alan Mathison Turing (Paddington, Londres, 23 de junio de 1912-Cheshire, 7 de junio de 1954), fue un matemático, lógico, científico de la computación, criptógrafo, filósofo, maratoniano y corredor de ultra distancia británico. Considerado uno de los padres de la ciencia de la computación y precursor de la informática moderna. Proporciono una influyente formalización de los conceptos de algoritmo y computación, LA MAQUINA DE TURING.

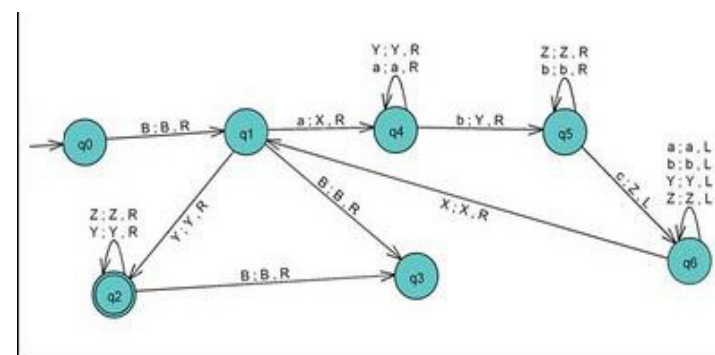


Figura 4: Ejemplo de una Máquina de Turing

Una máquina de Turing es un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo a una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es particularmente útil en la explicación de las funciones de una CPU dentro de un computador. (Feinman Richard, 2008)

Definición formal:

Una máquina de Turing es un modelo computacional que realiza una lectura/escritura de manera automática sobre una entrada llamada cinta, generando una salida en esta misma. Una máquina de Turing con una sola cinta puede definirse como una 7-tupla:

$$M = (Q, \Sigma, \Gamma, s, b, F, \delta),$$

Donde:

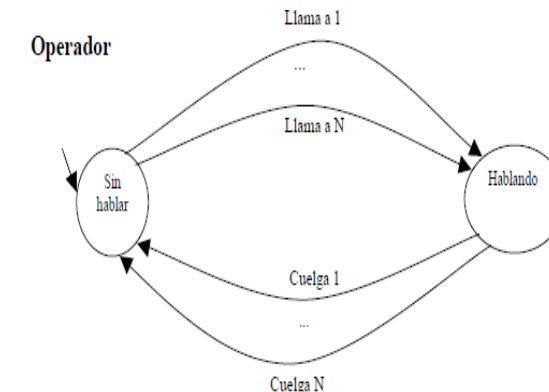
- $\{Q\}$ es un conjunto finito de estados.
- Σ es un conjunto finito de símbolos distinto del espacio en blanco, denominado alfabeto de máquina o de entrada.
- Γ es un conjunto finito de símbolos de cinta, denominado alfabeto de cinta ($\Sigma \subseteq \Gamma$).
- $s \in Q$ es el estado inicial.
- $b \in \Gamma$ es un símbolo denominado blanco, y es el único símbolo que se puede repetir un número infinito de veces.
- $F \subseteq Q$ es el conjunto de estados finales de aceptación.
- $\delta : Q \times \Sigma \rightarrow Q \times \Gamma \times \{L, R\}$ es una función parcial denominada función de transición, donde L es un movimiento a la izquierda y R es el movimiento a la derecha. (Feinman Richard, 2008)

Existen en la literatura un abundante número de definiciones alternativas, pero todas ellas tienen el mismo poder computacional, por ejemplo se puede añadir el símbolo S como símbolo de “no movimiento” en un paso de cómputo.

En los siguientes gráficos se muestra como resultado algunos ejemplos de los autómatas de estados finitos:

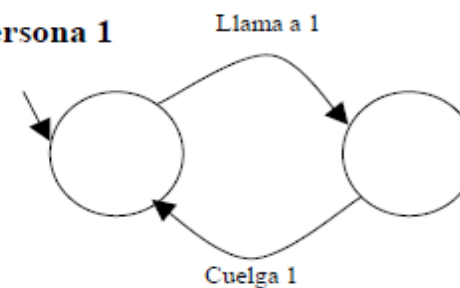
Ejemplo 1: Comportamiento de un operador telefónico:

- Modelo de un operador telefónico que llamar a tres personas diferentes, pero sólo a una de estas a la vez.

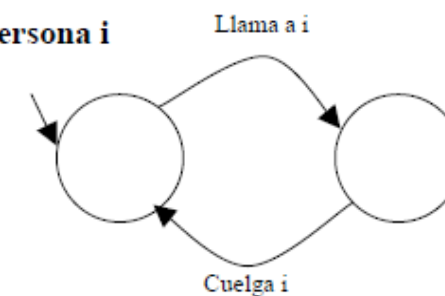


- Si en lugar de tres personas, las mismas son 100, o 1000, o 10000, rápidamente nos encontramos con lo siguiente:

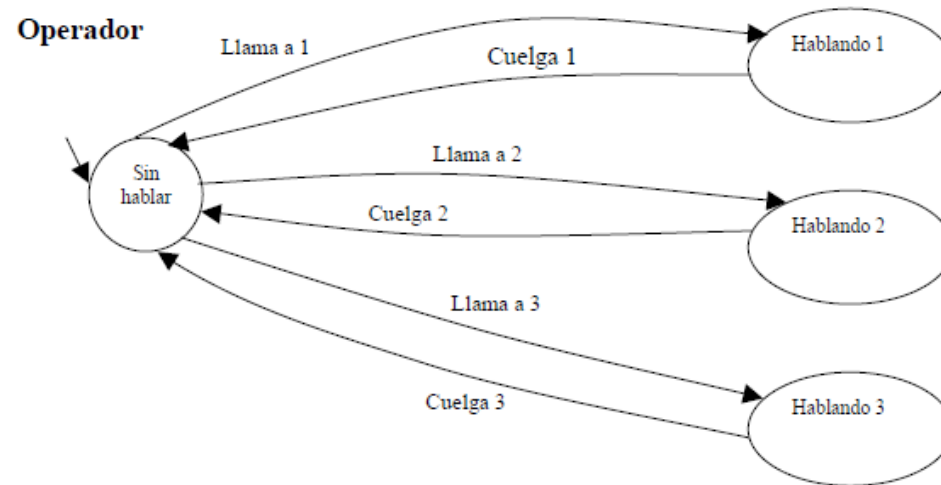
Persona 1



Persona i

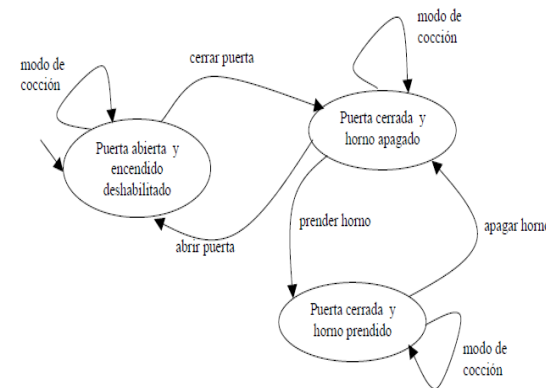


- Es decir, identificar en el operador, no sólo si está hablando o no, sino que también con quien lo está haciendo (en el otro ejemplo, esta información es deducible).



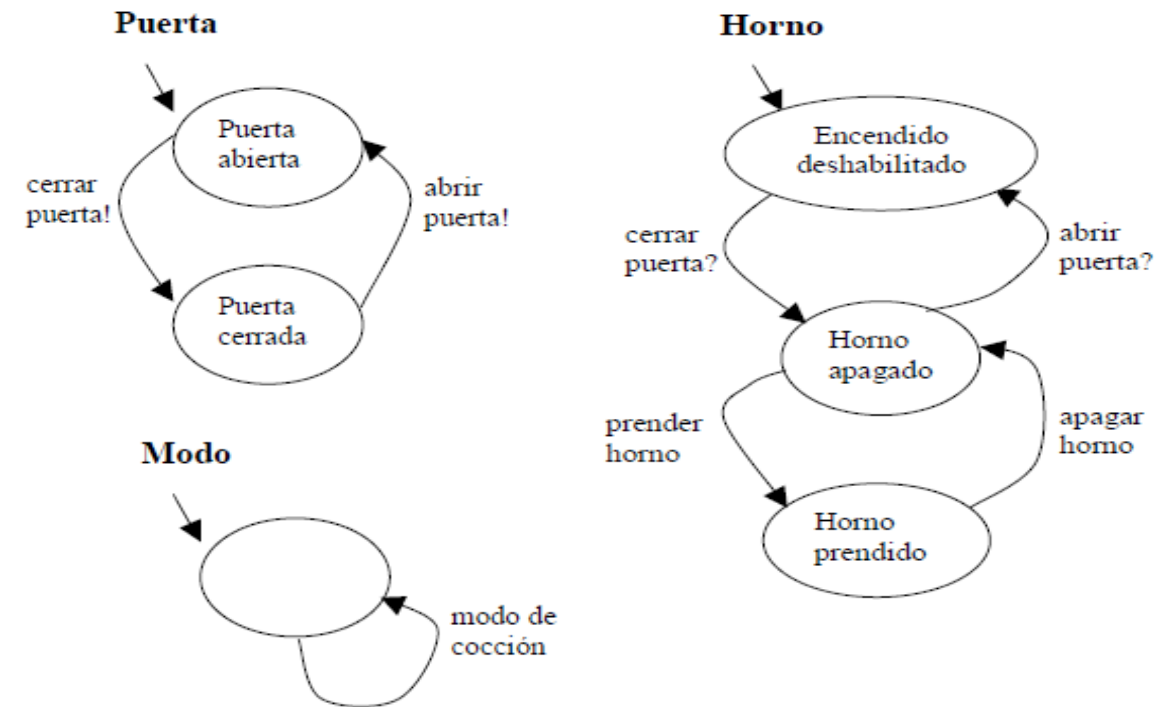
Ejemplo 2: Comportamiento de un horno microondas:

- El horno microondas posee una puerta. Si la puerta está cerrada, entonces puede estar o no en funcionamiento (según se prenda o apague). Estando prendido no es posible abrir la puerta del horno sin antes apagarlo. También asumamos lo siguiente: en cualquier momento es posible establecer el modo de cocción.



- Podemos descomponer este problema en tres partes: por un lado

modelaremos una puerta que se abre y se cierra, y por otro un horno que se prende y apaga y trataremos el modo de cocción de manera independiente.



Las máquinas de estados finitos se han ido aplicando en diferentes áreas de la informática por tanto, han llegado a diferentes sectores de la industria.

Aunque el conocimiento sobre estas máquinas de estados sea amplio y el paradigma de autómatas esté presente en numerosos proyectos, los usos reales de las variantes (conocidas a nivel teórico) o extensiones de una máquina con carácter determinista son prácticamente inexistentes en la industria del desarrollo de software.

Un claro ejemplo, los cuales parece generar bastante controversia debido a su estructura pobre en términos de IA (Inteligencia artificial), basado en máquinas de estados y que se utiliza en este tipo de proyectos son numerosos al menos el uso de mecanismos que implícitamente se comportan como tal.

Conclusiones.

Las Maquinas de estados Finitos, nos sirven para realizar procesos bien definidos en un tiempo discreto. Reciben una entrada, hacen un proceso y nos entregan una salida. Notemos que estas máquinas hacen una computación.

Por esta misma razón es que a lo largo de la investigación les presentamos una variedad de máquinas que tienen mucha relación y que pueden resolver un estado finito.

Por lo tanto una computación es capaz de resolver un problema, sí y solo sí tiene una solución algorítmica, es decir, puede ser descrito mediante una secuencia finita de pasos bien definidos un claro ejemplo es la máquina de Turing.

Podemos decir que cualquier solución a un problema que pueda ser representada por una Maquina de Turing está haciendo una computación. Si una máquina imprime dos tipos de símbolos, donde el primer tipo (llamadas figuras) consiste íntegramente en 1 y 0 y el segundo tipo serán llamados símbolos de la segunda especie entonces podemos llamar a esa máquina computadora.

Referencias.

Referencias de publicaciones periódicas:

- MINSKY (1967). Marvin L. Computation: Finite and Infinite Machines. Prentice-Hall.
- PABLO G. PADILLA. (2006) Lenguajes y álgebra de eventos regulares. Escuela Superior de Computo.
- TURING. A.M. (1936). On computable numbers, with an application to the Entscheidungsproblem London Math. Soc.
- TURING Alan. El pionero de la era de la información. España: Turner. ISBN 978-84-15832-12-6.
- API de la gestión de la IA basada en las máquinas de estados finitos en c#.
- MOORE Edward. Memorial Resolution of the Faculty of the University of

Wisconsin-Madison on the Death of Professor Edward Moore (PDF file).

- TURING Alan. En el Mathematics Genealogy Project. TFM Jose Alapont Lujan.(PDF file).
- FEINMAN Richard. (2008). Wikipedia. Obtenido de Wikipedia: es.wikipedia.org/wiki/Máquina_de_Turing
- TRAKHTENBROT B.A. (1973). Algoritmos y Computadoras. Limusa,
- JONES J. (1991). Herramientas informáticas. Prebntica Hall.
- HOPCROFT (1969)and ULLMAN. Formal Languages and their Relation to Automata. Addison-Wesley.
- MOTWANI(2001) HOPCROFT and ULLMAN. Introduction to Automata Theory,language and computacion,pearson Education.
- RICHARD JHONSONBAUGH.(1988) Matematicas Discretas. Iberoam´erica.
- LAWSON Mark V. Finite Automata. (2004)Chapman and Hall-CRC.

Sitiso Web

- SlideShare. (28 de octubre de 2010). Obtenido de <http://es.slideshare.net/junito86/presentacion-2-maquinas-de-estado-finito>
- Wikipedia. (s.f.). Obtenido de https://es.wikipedia.org/wiki/M%C3%A1quina_s%C3%ADncrona
- Wikipedia.org. (13 de octubre de 2016). Obtenido de wikipedia.org: https://es.wikipedia.org/wiki/M%C3%A1quina_de_Mealy

Artículo Recibido: 5-09-2016;

Artículo Aceptado: 31-10-2016

Máquinas de Turing en el Arte Musical

Turing Machine in Musical Art

Andres Antonio Choque Diaz

tonny.tonino@gmail.com

Carrera de Ingeniería de Sistemas -Universidad La Salle

Resumen

Una máquina de Turing principalmente es un dispositivo que transforma un INPUT en un OUTPUT después de algunos pasos y que pasa a través de una cinta infinita en la cual se va procesando los estados continuamente. Lo cual es un concepto similar a la música, donde podemos interpretar el pentagrama o la tablatura como la cinta, las notas como los estados y aquella secuencia lógica infinita como la melodía de una canción. Aunque el mundo musical es vasto y su medio de relación son los instrumentos musicales, la guitarra es el medio por el cual se interpretara la práctica del estudio.

La finalidad de esta investigación es comprobar la formulación de una estructura que permita combinar las máquinas de Turing junto con el lenguaje musical, la cual se justifica como una manera de integración y de aplicación de los conocimientos sobre autómatas. Los resultados son una estructura que comprendiendo de algunas restricciones nos retorna una relación entre estos dos conceptos ya mencionados.

Palabras claves
Máquina, Turing, lenguaje, interpretación, música, melodía, estados.

Abstract

A Turing machine is mainly a device that transforms an INPUT OUTPUT after a few steps and pass through an endless tape which is continuously processing states. Which is a concept similar to music, which we can interpret the staff or tablature as tape, notes how states and that infinite logical sequence as the melody of a song. Although the musical world is vast and its means of connection are the musical instruments, the guitar is the object by which the practice of the study were interpreted.

The purpose of this research is to test the formulation of a structure that combine turing machines along with the musical language, which is justified as a way of integration and application of knowledge about automata. The results are a structure comprising some restrictions we returned for a relationship between these two concepts already mentioned.

Key words

Machine, Turing, language, interpretation, music, melody, states.

Introducción

Alan Turing, creo lo que se llama como la máquina de Turing, lo cual es un dispositivo de reconocimientos de lenguajes regulares. Es más general que cualquier autómata finito y cualquier autómata de pila, debido a que ellas pueden reconocer tanto los lenguajes regulares, como los lenguajes independientes de contexto y además muchos otros tipos de lenguajes [1].

Actualmente la música tiende hacia la utilización de instrumentos y tecnología musical electrónica, por lo cual cada vez llega a unirse más hacia el mundo informático. Habiendo comprendido estos hechos, podemos interpretar la música y el lenguaje musical como un autómata que puede pasarse a través de esta máquina y así poder automatizarse y convertirse en un conjunto de algoritmos.

El objetivo de la investigación es el de exponer y demostrar que es posible relacionar al lenguaje musical y toda su estructura con las máquinas de Turing para su automatización

Referentes conceptuales.

La máquina de Turing (abreviado MT) tiene, un control finito, una cabeza lectora y una cinta donde puede haber caracteres, y donde eventualmente viene la palabra de entrada. La cinta es de longitud infinita hacia la derecha, hacia donde se extiende indefinidamente, llenándose los espacios con el carácter blanco (que representaremos con “t”). La cinta no es infinita hacia la izquierda, por lo que hay un cuadro de la cinta que es el extremo izquierdo, la MT la cabeza lectora es de lectura y escritura, por lo que la cinta puede ser modificada en curso de ejecución. Además, en la MT la cabeza se mueve bidireccionalmente (izquierda y derecha), por lo que puede pasar repetidas veces sobre un mismo segmento de la cinta [2].

Cabe resaltar que se debe comprender completamente el funcionamiento de la máquina para así poder comprender los puntos que este estudio desea presentar.

Ahora, podemos bien detallar que la máquina de Turing contiene principalmente un conjunto de características. Posee un lenguaje que será utilizado para ser procesado. También posee una cinta en la cual se pasa de diferentes estados del lenguaje, la cual es infinita hacia la derecha y que está siempre pasa a través de una caja y un cabezal que va procesando los estados.

La música a su vez, es por definición el arte de organizar sensible y lógicamente una combinación coherente de sonidos y silencios utilizando los principios fundamentales de la melodía, la armonía y el ritmo, mediante la intervención de complejos procesos psico-anímicos [3].

Estos tres conceptos son fundamentales en su análisis. Melodía es la sucesión de sonidos que es percibida como una sola identidad [4]. Armonía es el equilibrio entre las distintas partes de un todo y su resultado connota “Belleza” [5]. Y por

último el ritmo son los movimientos marcados por la sucesión regular de elementos débiles y fuertes, o bien de condiciones opuestas o diferentes [6].

Enfatizamos que en este artículo se analizara a la música como un “Proceso mecánico posible a ser automatizado”.

Musica=Armonia+Melodia+Ritmo

(1.1)

Principalmente la música puede ser vista como la suma de las partes involucradas. Sin embargo, nuestro concepto (1.1) es una visión general de esta, pero lo que nosotros debemos analizar es la base de la formulación que es el lenguaje musical.

$V=\{ \lambda, c, c\#, d, d\#, e, f, f\#, g, g\#, a, a\#, b\}$

(1.2)

donde: nota(i)sostenido

equivale a nota(i+1)bemol

Bajo esta determinación del lenguaje, podemos comenzar a interpretarlo en base al proceso de la máquina de Turing.

Ahora, para que este lenguaje pueda ser analizado en la MT, es necesario tener una cinta, la cual en lenguaje musical pueden ser varios y todos tan útiles como los otros.

El pentagrama, o pauta musical, es el lugar donde se escriben las notas y todos los demás signos musicales en el sistema de notación musical occidental [7].

Y por otra parte tenemos la Tabladura, se utiliza para definir formas de escritura musical especiales para ciertos instrumentos y que, a diferencia de la notación musical corriente, presentan únicamente las posiciones y colocaciones en el instrumento para la interpretación de una pieza, y no las alturas ni las duraciones de los tonos [8].

La guitarra, también conocida como guitarra clásica o guitarra española,1 es

un instrumento musical de cuerda pulsada, compuesto de una caja de madera, un mástil sobre el que va adosado el diapasón o trastero —generalmente con un agujero acústico en el centro de la tapa (boca)—, y seis cuerdas. Sobre el diapasón van incrustados los trastes, que permiten las diferentes notas [9].

Nuestro estudio se basará en la guitarra de 12 trastes y el análisis de los acordes de esta.

escala mayor(nota)= nota(i)+nota(i*2)

(1.3)

La figura 1.3 nos entrega una escala mayor recorriendo desde la primera palabra del lenguaje hasta la última. Y debe ser cíclica, desde la palabra del lenguaje a utilizar hasta retornar a una menos que esta [10].

Posteriormente en base a esta se forman los acordes que son conjuntos de notas bajo los principios de la música.

acorde(nota)= notas(i)+notas(i+3)+noas(i+5)

acorde=melodia+armonia

(1.4)

Aunque hay otras variaciones de los acordes, nosotros estudiaremos como intención de estudio a la tónica mayor con los acordes mayores.

Ahora podemos analizar a través de la MT tomando la base una linealidad donde estas notas atraviesan.

Y esta máquina los analiza, a través de los procesos entregados. Pero para poder realizar correctamente esto, se deben agregar otras características.

“A mayor tiempo menor armonia”

“La secuencia de notas debe ser
melodica y armonica”

“El ritmo depende del tiempo”

(1.5)

La MT debe poseer las características dadas en 1.5, lo cual dificulta más la utilización. Cabe resaltar que hay algunos estereotipos y paradigmas que deben aclararse para este estudio:

Una canción que es un segmento de música, puede ser infinita, pero al no ser humanamente posible, se limita a un determinado tiempo. Por otra parte, al aumentar el tiempo la armonía que es un concepto humano, tiende a disminuir, por lo que no técnicamente no es válida una duración infinita, pero si es válida teóricamente.

Posteriormente no podemos secuenciar notas aleatoriamente, por lo que debemos respetar hechos melódicos y armónicos. Y tampoco podemos poner notas secuencialmente ya que se deben respetar reglas de tiempo.

Al ser la MT una representación, también para fines técnicos, vamos a representar la interpretación con una depreciación de melodía y armonía, pero si vamos a considerar el ritmo que es importante y lo más tangible en el estudio.

1		G		
2		B		
3		D	E	C#
4	C#	G	G	
5		D		
6		G		

Tabla 1. Ejemplo de secuencia en tablatura.

Como se puede observar en la tabla 1 podemos secuenciar los elementos del lenguaje en una tablatura que nos representa cada línea de la guitarra en un determinado tiempo.

Entonces, para poder representar estas notas, debemos pasar por un proceso que es conocido como rasgueo y punteo.

punteo=cuerda

rasgueo= $\sum$ cuerdas

(1.6)

Entonces teniendo claro este concepto, podemos determinar que la acción que la MT va a realizar es la de puntear o rasguear para así pasar al próximo estado. Y también la de elegir rítmicamente los siguientes elementos en la cinta (tablatura) a utilizar.

1= nota= palabra de lenguaje

0=silencio= λ

El ritmo podrá obtener 2 estados que son con una palabra del lenguaje determinada o λ , lo cual representa un silencio, si se desea un silencio prolongado se debe establecer los λ seguidamente de cuanto se desee.

En la interpretación la cinta que atraviesa la MT se podrá interpretar como 6 cintas que son la representación de la tablatura, las cuales están relacionadas entre si ya que poseen una correlación en el sonido, el lenguaje y el tiempo de proceso de los estados.

Entonces el proceso será el siguiente: se procede a la toma del primer estado que es la nota o el conjunto de notas en el pentagrama, el cual interpretado por la maquina determinara si debe ser rasgueado o punteado. Posteriormente una vez concluido este análisis del primer proceso, se prosigue con el siguiente y se procede a hacer el mismo análisis infinitas veces o finitas, dependiendo de la duración.

Esta interpretación de la maquina es factibles tomando en cuenta las suposiciones y advertencias dadas anteriormente en el planteamiento del problema.

Actualmente haciendo uso de la programación computacional, es posible lograr este cometido.

Lego Mindstorm [11] es un proyecto de la compañía lego para poder realizar las interpretaciones que necesitemos.

Y utilizando materiales es posible recrear es posible realizar la interpretación de las MT sobre una guitarra y así comenzar el proceso de construcción.



Imagen de fastythefastcat [12] [13]

Métodos.

El método de la observación científica, consiste en la percepción directa del objeto de investigación. La observación investigativa es el instrumento universal del científico. La observación permite conocer la realidad mediante la percepción directa de los objetos y fenómenos.

Método deductivo que primero consiste en encontrar principios desconocidos, a partir de los conocidos. Una ley o principio puede reducirse a otra más general que la incluya. Si un cuerpo cae decimos que pesa porque es un caso particular de la gravitación.

También sirve para descubrir consecuencias desconocidas, de principios conocidos. Si sabemos que la formula de la velocidad es $v=e/t$, podremos calcular la velocidad de un avión. La matemática es la ciencia deductiva por excelencia; parte de axiomas y definiciones.

Entre los métodos que se utilizaron en la construcción de este trabajo, están el método observativo y deductivo, por los cuales se dio la idea e intención del estudio, y debido a estos métodos este proyecto esta sujeta a varias restricciones. Mientras más restricciones se presenten, el proyecto mayor deductividad presenta. El método no experimental está vigente ya que este estudio es un establecimiento teórico, por lo cual no es posible (aun) su total aplicación práctica.

Resultados y discusión

El estudio presentado, es principalmente teórico y sin ningún antecedente de proyectos anteriores, aunque podríamos citar las cajas musicales como un ejemplo que provee facilidad de abstracción, el estudio llega a ser mucho mas complejo que esto. Aunque se de por intención relacionar el lenguaje musical hacia un mundo cuantificable y medurable, no se pueden proveer medios estadísticos o cuantificablemente analíticos por el momento. Es decir hasta que este estudio sea experimentado o simulado.

Se deben tomar muy en cuenta las restricciones del proyecto, ya que varias de estas nacieron de la necesidad de que el hombre aun no cuenta con esta tecnología, pero el planteamiento de estas restricciones fueron necesarias para la estructura de la investigación.

Conclusiones:

Podemos concluir que, a través de la utilización de MT, es posible representar lenguajes musicales con condiciones dadas. Actualmente no es posible que una maquina represente los fundamentos de melodía y armonía. Aunque podemos citar el concepto de "Fantasmas en la maquina" que explica que las maquinas presentan segmentos de aleatorios de códigos agrupados que dan protocolos inesperados, lo que estos radicales libres generan dudas sobre el libre albedrio en estas máquinas, estos son conceptos que no pueden ser demostrados científicamente y por lo tanto tampoco en la MT que solo sirve para demostrar conceptos claros y exactos.

Cabe también resaltar que este estudio no es una interpretación general de la música y no busca menospreciar el potencial humano al buscar automatizar esta. La música es un arte abstracto, y aunque tienda a ser más "electrónico" aún hay puntos y conceptos que no podemos llegar a completar con las máquinas. Las máquinas de Turing pueden analizar el lenguaje musical parcialmente, pero aun así se mantiene una dependencia del potencial humano

Referencias Web

[1] <http://maquinaturing.blogspot.com/p/funcionamiento-de-la-maquina-turing.html>

[2] <http://maquinasdeturing.blogspot.com/2010/08/3-que-es-una-maquina-de-turing-y-como.html>

[3] <https://es.wikipedia.org/wiki/M%C3%BAsica>

[4] <https://es.wikipedia.org/wiki/Melod%C3%ADa>

[5] <https://es.wikipedia.org/wiki/Armon%C3%ADa>

[6] <https://es.wikipedia.org/wiki/Ritmo>

[7] <https://es.wikipedia.org/wiki/Pentagrama>

[8] <https://es.wikipedia.org/wiki/Tablatura>

[9] <https://es.wikipedia.org/wiki/Guitarra>

[10] <http://elclubdelautodidacta.es/wp/2011/12/construccion-de-acordes-1/>

[11] <https://education.lego.com/en-au/learn/middle-school/mindstorms-ev3/teaching-resources/software/tablet-app>

[12] https://es.wikipedia.org/wiki/Lego_Mindstorms

[13] <https://www.youtube.com/watch?v=cXgB3IlvPHI>

Artículo Recibido: 11-09-2016
Artículo Aceptado: 24-10-2016

ILUMINATE

**REVISTA DE INVESTIGACION DE LA CARRERA
DE INGENIERÍA DE SISTEMAS**

UNIVERSIDAD LA SALLE EN BOLIVIA

Misión

Nuestra Misión es publicar, divulgar trabajos de investigación generados en el ámbito académico de Las Ciencias de la Computación a nivel nacional e internacional, elaborado principalmente por estudiantes e investigadores para aportar a la producción científica de nuestra universidad y de nuestro país.

Política Editorial

Illuminate, es una publicación editada por la Carrera de Ingeniería de Sistemas de la Universidad La Salle de Bolivia, con frecuencia de publicación anual, busca divulgar trabajos de investigación del area de las ciencias de la computación generados en el ambiente Universitario por estudiantes y en entornos colaborativos con otras universidades. Se reciben contribuciones en español e inglés. El Artículo candidato a publicación debe cumplir con las normas que aparecen en las instrucciones para los autores. Luego de su recepción, el artículo se somete a evaluación por pares, los que recomiendan su aceptación o rechazo.

La revista solicita a los autores con preferencia remitir artículos originales u originales cortos, artículos inéditos en español o inglés con carácter científico, que serán valorados por el comité editor.

La revista se divulga en forma impresa y electrónica, esta ultima para ampliar el espectro de difusión, nos interesa que profesionales, pequeñas, medianas y grandes empresas, organismos, tomadores de decisiones, otros investigadores, docentes y estudiantes universitarios, puedan acceder a sus páginas y establecer referencias para potenciar el desarrollo local y regional.

Licenciamiento

Los artículos que sean seleccionados para formar parte de la revista deberán ser autorizados por su autores o autores, para su difusión a través de cualquier medio, mediante formato establecido para dicho fin. Los artículos de la revista, permanecen en acceso abierto disponibles en la página web institucional como en los diferentes índices y bases de datos a los cuales la revista está suscrita.

Instrucciones para la presentación de Artículos

1. La entrega de artículos y afiliación

Los artículos, serán recibidos y registrados en Dirección Académica de la Universidad La Salle:

Editor en Jefe - Revista Illuminate

wpenafiel@ulasalle.edu.bo;

Instituto de Investigaciones, Universidad La Salle

Calle Jorge Carrasco, esquina Las Palmas, No. 450, zona Bolognia-La Paz, Bolivia

Tel: 591-2-2723588 - 2723598 / Fax: 591-2-2723588

<http://www.ulasalle.edu.bo/Illuminate>

La universidad entregará, en este caso, acuse de recepción por la misma vía.

Se entregarán en formato digital en una versión actualizada de Word. Las fechas de recepción y aceptación del artículo son importantes, y deberán ocupar un lugar en el artículo de acuerdo con el formato para la presentación de artículos.

En caso de ser el primer aporte de la revista, el autor deberá presentar en formato digital, nombres y apellidos completos, grado académico, lugar y fecha de nacimiento, lugar de trabajo actual, cargo actual y dirección correo electrónico, para su respectiva afiliación.

Los autores publicados en Illuminate se afiliarán automáticamente a la publicación para recibir periódicamente la revista en formato impreso.

2. Proceso de arbitraje

Una vez recibido el artículo y que se ha comprobado que cumple con los requisitos de forma indicados en las instrucciones para el autor, se acusará recibo mediante correo electrónico y se iniciará el proceso de selección.

Los revisores evaluadores serán designados por el Comité Editorial. Se empleará la metodología “revisores ciegos”, quienes evaluarán los aspectos propios de la especialidad tratada.

Los artículos recibidos y admitidos por el Comité Editorial serán enviados, con la información de los autores a dos (2) evaluadores de reconocido prestigio en el área y materia del artículo remitido, también se remitirá el formulario de evaluación.

Los evaluadores tendrán cuatro semanas para revisar el artículo y entregar el formulario de evaluación lleno. Los árbitros evaluadores pueden sugerir modificaciones al artículo original que deben ser realizados por el autor para la aceptación de su artículo. El autor tiene (15) días hábiles para entregar el artículo reformulado. En caso de existir diferencia de opinión entre los dos árbitros evaluadores, se procederá a enviar el artículo a un tercer evaluador o al Editor que dirimirá la controversia.

La evaluación negativa se podrá justificar en los siguientes casos:

- a) La clasificación propuesta no concuerda con el desarrollo del artículo.
- b) Si presenta deficiencias en la redacción o el desarrollo científico.
- c) Si la información contenida no respeta los derechos de autor.
- d) Si el artículo no corresponde al eje temático de la revista.

3. Conflicto de Interés

Illuminate aplicará mecanismos internos de resguardo y cautela, para evitar conflicto de interés que afecte a la objetividad en todo el proceso editorial y que involucren al comité editorial y a los evaluadores anónimos en el proceso de arbitraje.

4. Aceptación de artículos

Una vez concluida la etapa de evaluación, será comunicada al autor la decisión sobre la aceptación o rechazo del artículo por parte del Comité Editorial.

El artículo deberá presentar la fecha de entrega y aceptación, éstas serán impresas de acuerdo con las instrucciones para los autores. Adicionalmente

el autor deberá llenar el formulario de autorización para la publicación del artículo o revista escrita y/o digital

El comité editorial, declara que las opiniones y comentarios expresados en los artículos son de responsabilidad exclusiva de sus autores.

La revista Illuminate comunicará al autor, el informe final de evaluación. Dicha Comunicación tendrá lugar en un plazo máximo de seis (6) meses, a contar desde la fecha de recepción del artículo.

5. Publicación

El artículo aceptado será publicado en el siguiente número de revista disponible de espacio para el efecto.

6. Afiliación de Evaluadores

Un árbitro evaluador será parte de la base de datos de Illuminate al momento de ser nombrado por el comité editorial y haber aceptado la primera solicitud de evaluación de un artículo. Para su registros será necesario ingresar el curriculum vitae del evaluador con los siguientes datos: nombres y apellidos completos, lugar de estudio, y dirección correo electrónico.

7. Conformación del Comité Editorial

El comité editorial será conformado por disposición de Consejo de la carrera de Ing. de Sistemas de la Universidad La Salle en Bolivia, nombrando para el efecto los siguientes estamentos; Editor, Editor responsable, Consejo editorial, Consejo de Arbitraje, Colaboradores, Diseño y Diagramación.

El Consejo Editorial aprobará la incorporación de nuevos evaluadores.

8. Compromisos y Responsabilidades éticas

Cada autor presentará una declaración responsable de autoría y originalidad respecto de los siguientes aspectos:

- Inédito. No se acepta material previamente publicado. Los autores son responsables de obtener los oportunos permisos para reproducir parcialmente (texto, tablas o figuras) de otras publicaciones y de citar su procedencia correctamente. Las opiniones expresadas en los artículos publicados son responsabilidad del autor/es
- Autoría. En la lista de autores firmantes deben figurar únicamente aquellas

personas que han contribuido intelectualmente al desarrollo del trabajo. Haber colaborado en la recolección de datos no es, por si mismo, criterio suficiente de autoría. Illuminate declina cualquier responsabilidad sobre posibles conflictos derivados de la autoría de los trabajos que se publiquen.

- Consentimiento informado. Los autores deben mencionar que los procedimientos empleados en sujetos de experimentación han sido realizados con consentimiento informado.

Instrucciones para los autores

Todos los artículos serán publicados en español, con título en español e inglés, con resúmenes en español e inglés.

1. Instrucciones respecto al tipo de artículo

Los trabajos se estructurarán de acuerdo con las secciones y normas que a continuación se exponen:

- a) Artículos originales (investigación científica). Documento que presenta, de manera detallada, los resultados originales de proyectos terminados de investigación.

Los artículos originales, de reporte de caso y cortos, deben cumplir con la siguiente estructura: la introducción debe resaltar la importancia de la investigación, presentar la literatura relacionada y entregar antecedentes necesarios para comprender la hipótesis de los autores, terminando con un párrafo que indique claramente los objetivos de la investigación. El Método debe tener suficiente información que permita a otro investigador replicar el ensayo y lograr los mismos resultados, así como la inclusión del diseño experimental, el análisis estadístico y las referencias de los métodos ya publicados.

Los resultados y discusión se deben presentar en forma clara, apoyados con cuadros y figuras, con el análisis estadístico y los antecedentes de otros investigadores. Las conclusiones deben redactarse de acuerdo con los objetivos de la investigación explicando claramente los principales resultados de la investigación. Las referencias deben contener todos los documentos consultados.

b) Artículo de reflexión. Documento que presenta resultados de investigación terminada desde una perspectiva analítica, interpretativa o crítica del autor, sobre un tema específico, recurriendo a fuentes originales. Los artículos de reflexión deben mostrar: a) intención analítica; b) propósito interpretativo; c) posición crítica. La estructura: Resumen, Introducción, Desarrollo del tema, Conclusiones y referencias.

c) Artículo de revisión. Documento resultado de una investigación terminada donde se analizan, sistematizan e integran los resultados de investigaciones publicadas o no publicadas, sobre un campo en ciencia o tecnología, con el fin de dar cuenta de los avances y las tendencias de desarrollo. Se caracteriza por presentar una cuidadosa revisión bibliográfica de por lo menos 40 referencias. Los artículos de revisión deben tener: Resumen, Introducción, Desarrollo del tema, Conclusiones y referencias.

d) Artículo original corto. Documento breve que presenta resultados originales preliminares o parciales de una investigación científica o tecnológica, que por lo general requieren de una pronta difusión. Tiene la misma estructura de los artículos originales.

e) Reporte de caso. Documento que presenta los resultados de un estudio sobre una situación particular con el fin de dar a conocer las experiencias técnicas y metodológicas consideradas en un caso específico. Incluye una revisión sistemática comentada de la literatura sobre casos análogos.

f) Revisión de tema. Documento resultado de la revisión crítica de la literatura sobre un tema en particular.

g) Cartas al editor. Posiciones críticas, analíticas o interpretativas sobre los documentos publicados en la revista, que a juicio del Comité Editorial constituyen un aporte importante a la discusión del tema por parte de la comunidad científica de referencia.

h) Editorial. Documento escrito por el editor, un miembro del comité editorial o un investigador invitado sobre orientaciones en el dominio temático de la revista.

i) Traducción. Traducciones de textos clásicos o de actualidad o transcripciones de documentos históricos o de interés particular en el dominio de publicación de la revista.

j) Documento de reflexión no derivado de investigación. Reflexión sobre un tema en particular, el cual no requiere ser derivado de una investigación científica o tecnológica

k) Reseña bibliográfica.

l) Otros

En todos los casos el artículo debe presentar aspectos de originalidad y contener las referencias bibliográficas.

Adicionalmente el autor deberá llenar el formulario de autorización para la publicación de artículos y/o revistas en formatos impresos y digitales.

2. Instrucciones respecto al formato del artículo

Los artículos seleccionados serán el centro de la revista, por lo que deben ajustarse a las siguientes especificaciones de forma:

a) Título del artículo (en español e inglés) no debe tener más de 15 palabras. El título estará escrito con letras minúsculas y en negritas con tipo Times New Román de 12 puntos y centrado, este debe contener máximo 15 palabras. (En español e inglés).

b) Lista de Autores: Con los nombres completos y apellidos en el orden que deben aparecer. Adicionalmente el autor principal anotará su correo electrónico, para la correspondencia de editores y lectores; lugar y fecha de investigación y financiamiento si la tuvo.

c) Resumen y palabras clave: Se presenta un resumen en español y en inglés máximo de 100 palabras. Palabras clave en español e inglés (hasta un máximo de diez palabras).

d) Texto del artículo: Cuando los artículos son originales y original corto

deberán tener las siguiente partes:

Introducción: El autor debe establecer el propósito del estudios, resumir su fundamento lógico.

Materiales y métodos: Incluye la selección de procedimientos para el trabajo experimental, y se indentifican los métodos y equipos con suficiente detalle.

Resultados y Discusión: Debe presentarse en secuencia lógica con sus re4spectivas tablas y gráficas y los comentarios de los principales hallazgos. En la Discusión enfatizar los aspectos más importantes del estudio.

Conclusiones: Anotar una o más conclusiones

Los demás artículos deberán tener como mínimo resúmen, palabras clave, el desarrollo del tema y las conclusiones.

- e) El artículo debe estar capturado electrónicamente en una versión actualizada de Word, su extensión no debe exceder las doce (15) páginas en una columna y espacio sencillo, letras Times New Roman de 10 puntos, en hojas tamaño carta.
- f) Los títulos principales deberán siempre ser precedidos de una línea en blanco, salvo que con ellos se comience una nueva página. Los títulos principales se escribirán centrados utilizando letras mayúsculas y en negritas con tipo Times New Román de 10 puntos. Se dejará una línea en blanco después de cada título principal. Cada título tendrá al menos dos líneas de texto que le continúen al final de cada hoja.
- g) Las tablas deberán intercalar en el texto, ubicándose lo más cerca posible a la sección de texto que las refiere, pero no será aceptable, por fines de uniformidad, dividir la página en varias columnas para intercalar tabla en el texto. Se deberán centrar con respeto al formato del texto y deberá existir por lo menos una línea en blanco antes y después de cada tabla. La numeración de tablas deberá hacerse de forma secuencial. Los títulos de las tablas deberán ir en su parte inferior.

h) Las figuras se deberán intercalar con el texto, ubicándose lo más cerca posible a la sección de texto que las refiere, pero no será aceptable, por fines de uniformidad, dividir la página en varias columnas para intercalar figuras con el texto. Se deberán centrar con respecto al formato del texto y deberá existir por lo menos una línea en blanco antes y después de cada figura. La numeración de figuras deberá hacerse de forma secuencial. Los títulos tendrán como máximo una extensión de dos líneas. Las figuras no deberán enmarcarse, salvo que se trate de gráficas donde esto último se justifique. Asimismo, en la instrucción de formato de imagen de Word, deberá asegurarse que la figura se defina como un objeto en línea con el texto, con la finalidad de que el control de la posición de la imagen en el mismo sea fácil de manipular por los editores en caso que se requiera algún ajuste menor en el artículo.

- i) Si desea incluir una sección de agradecimientos, esta se ubicará después de la sección de conclusiones y antes de la sección de referencias, utilizando el mismo formato que para títulos y secciones principales.
- j) Las fuentes bibliográficas deben ser citadas a lo largo del texto, en formato ISO.
- k) Cualquier tema no cubierto en lo anterior será resuelto por el editor coordinador en el Consejo editorial y deberán ser consultados al siguiente correo electrónico: wpenafiel@ulasalle.edu.bo



CAMPUS BOLOGNIA:

Av. Jorge Carrasco esq.
Calle Las Palmas N° 450(Bologna)
Telfs.: (591-2) 2723588
(591-2) 2723598

e-mail: wpnafiel@ulasalle.edu.bo

[f/ulasallebolivia](https://www.facebook.com/ulasallebolivia)

www.ulasalle.edu.bo