UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA EL APRENDIZAJE DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU

CARRION ORDOÑEZ LUIS DAMIAN
INGENIERO DE SISTEMAS

MACHALA
2017

UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA EL APRENDIZAJE DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU

CARRION ORDOÑEZ LUIS DAMIAN
INGENIERO DE SISTEMAS

MACHALA
2017

UTMACH

UNIDAD ACADÉMICA DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

EXAMEN COMPLEXIVO

SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA EL APRENDIZAJE DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU

CARRION ORDOÑEZ LUIS DAMIAN
INGENIERO DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 21 DE AGOSTO DE 2017

MACHALA
21 de agosto de 2017

Nota de aceptación:

Quienes suscriben, en nuestra condición de evaluadores del trabajo de titulación denominado SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA EL APRENDIZAJE DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU , hacemos constar que luego de haber revisado el manuscrito del precitado trabajo, consideramos que reúne las condiciones académicas para continuar con la fase de evaluación correspondiente.

VALAREZO PARDO MILTON RAFAEL
0704518893
TUTOR - ESPECIALISTA 1

CÁRDENAS VILLAVICENCIO OSCAR EFRÉN
0703935312
ESPECIALISTA 2

HONORES TAPIA JOOFRE ANTONIO
0704811751
ESPECIALISTA 3

Fecha de impresión: lunes 14 de agosto de 2017 - 10:19

URKUND

Urkund Analysis Result

Analysed Document: TESIS DAMIAN.docx (D29674953)
Submitted: 2017-07-17 22:11:00
Submitted By: mvalarezo@utmachala.edu.ec
Significance: 2 %

Sources included in the report:

http://www.redalyc.org/pdf/2251/225125175002.pdf
http://ciep.itam.mx/~davidm/pub.htm

Instances where selected sources appear:

2

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, CARRION ORDOÑEZ LUIS DAMIAN, en calidad de autor del siguiente trabajo escrito titulado SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA EL APRENDIZAJE DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU , otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las dispociones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 21 de agosto de 2017

CARRION ORDOÑEZ LUIS DAMIAN
1103545198

DEDICATORIA

Dedico este proyecto a Dios por ser el inspirador para cada uno de mis pasos dados en mi convivir diario; a mis padres por ser los guías en el sendero de cada acto que realizo y ser el incentivo para seguir adelante con este objetivo.

Sr. Carrión Ordóñez Luis Damián

AGRADECIMIENTO

Agradezco profundamente a Dios, por guiarme en el sendero correcto de la vida e iluminarme en todo lo que realizo de mí convivir diario; a mis padres por ser mi ejemplo para seguir adelante en el convivir diario y por inculcarme valores que de una u otra forma me han servido en la vida; a mis maestros y compañeros de la Carrera de Ingeniería de Sistemas por compartir generosa y desinteresadamente durante este largo caminar universitario.

Sr. Carrión Ordóñez Luis Damián

RESUMEN

Dentro de la estructura de los computadores está la memoria RAM, cumple con la función de cargar todas las instrucciones que ejecuta unidad central de proceso (CPU), por ello el proceso de asignación de memoria es una tarea de suma importancia para el eficiente y eficaz funcionamiento de los programas.

Considerando lo anterior expuesto, mediante este proyecto se hará énfasis en el aprendizaje del funcionamiento interno de una de las partes más importantes de la computadora como es la memoria. Si bien es cierto existe una diversificación extensa donde se puede obtener dicho conocimiento sin embargo teniendo en cuenta las buenas prácticas de aprendizaje, se dice que un factor que facilita el aprendizaje es la visualización, por ello se realizó un programa de simulación de algoritmos de sustitución de páginas para el aprendizaje del funcionamiento interno de la técnica de memoria virtual FIFO y LRU.

Palabras claves: Algoritmos, memoria virtual, simulación, RAM, sistemas operativos.

ABSTRACT

Within the structure of the computers we have the RAM, which is responsible for loading all the instructions that Central Process Unit executes, so the process of allocating memory is a task of paramount importance for efficient and efficient operation.

Considering the above, through this project will emphasize the learning of the internal functioning of one of the most important parts of the computer such as RAM. Although it is true that there is extensive diversification where this knowledge can be obtained, however, taking into account good learning practices, it is said that a factor that facilitates learning is visualization, so a program of simulation of substitution algorithms Of pages for learning the internal operation of the virtual memory technique FIFO and LRU.

Keywords: Algorithms, virtual memory, simulation, RAM, operating systems.

CONTENIDO

Pág.

DEDICATORIA ... 6
AGRADECIMIENTO ... 7
RESUMEN ... 8
ABSTRACT ... 9
CONTENIDO ... 10
ILUSTRACIONES ... 12
1. INTRODUCCIÓN ... 13
1.1. Marco contextual ... 13
1.2. Problema ... 13
1.3. Objetivo General ... 13
2. DESARROLLO ... 14
2.1. Marco Teórico ... 14
2.1.1. Métodos actuales de aprendizaje ... 14
2.1.2. Entornos de simulación ... 14
2.1.3. Sistemas operativos ... 14
2.1.4. Memoria RAM: ... 14
2.1.5. Administración de memoria ... 15
2.1.6. Memoria virtual ... 15
2.1.7. Paginación de memoria ... 15
2.1.8. Fallos de página ... 16
2.1.9. Algoritmo ... 16
2.1.9.1. Algoritmo FIFO ... 16
2.1.9.2. Algoritmo LRU ... 16
2.2. Solución del problema ... 16
2.2.1. Materiales ... 16
2.2.2. Diagrama de flujo ... 17
2.2.3. Procedimiento ... 17
2.2.3.1. Procedimiento de algoritmo FIFO ... 18
2.2.3.2. Procedimiento de Algoritmo LRU ... 20

2.3. Resultados 22
2.3.1. Consideraciones teoricas 22
2.3.2. Consideraciones del diseño 22
2.3.3. Solución de la propuesta 22
3. CONCLUSIONES 23
REFERENCIAS BILIOGRÁFICAS 24
ANEXOS 26

ILUSTRACIONES

Ilustración 1: Esquema de traducción de la paginación. [16] 15
Ilustración 2: Datos para el algoritmo FIFO 18
Ilustración 3: Datos para el algoritmo LRU 20

Anexo 1: Diagrama de flujo de algoritmo FIFO 26
Anexo 2: Diagrama de flujo de algoritmo LRU 27

1. INTRODUCCIÓN

La tecnología avanza a pasos agigantados dirigiendo cada vez más las actividades que ocurren alrededor del mundo mediante herramientas, dispositivos o artículos que trabajan bajo diferentes aplicaciones, tal es el caso que el ser humano tiende a estar ligado directa o indirectamente con un ordenador [1] [2], sin embargo, un computador desde el punto de vista del hardware no es capaz de suplir con todas las necesidades por sí sola, por lo que se crean los sistemas operativos (SO) como una forma de comunicación, teniendo una total aceptación en nuestras vidas. Gracias a los avances tecnológicos también han surgido nuevos métodos de aprendizaje, entre uno de ellos se tiene los entornos de simulación [3], que logran el desarrollo cognitivo y metacognitivo de las personas [4].

1.1.Marco contextual

En la administración de memoria se consideran tres aspectos principales, debe ser virtual, compartido y distribuido [5]. Esto se logra mediante diversos sistemas de paginación, entre ellos FIFO y LRU. En este proyecto se expondrá estos sistemas aplicando un nuevo método de aprendizaje denominado Sistema de simulación [6] [7].

1.2.Problema

Así como los equipos informáticos cuentan con más capacidad de memoria, los programas también aumentan el uso de ella para ejecutarse, para gestionarla de manera óptima se ha dado paso a diversas técnicas tales como intercambio de memoria, paginación, segmentación, memoria virtual, estas han sido explicadas a un diverso número de alumnado, proporcionándoles libros, revistas, folletos, entre otros; por ello el problema no radica en la falta de información, sino más bien en el método de aprendizaje que se está utilizando, por lo que se considera hacer la SIMULACIÓN DE ALGORITMOS DE SUSTITUCIÓN DE PÁGINAS PARA APRENDIZAJE DEL FUNCIONAMIENTO INTERNO DE LA TÉCNICA DE MEMORIA VIRTUAL FIFO Y LRU.

1.3. Objetivo General

Crear un entorno de simulación de los algoritmos de sustitución de páginas para lograr un mejor aprendizaje del funcionamiento interno de la técnica de memoria virtual FIFO y LRU.

2. DESARROLLO

2.1. Marco Teórico

2.1.1. Métodos actuales de aprendizaje

La enseñanza en cualquier índole tiene como fin el aprendizaje eficiente y eficaz del alumnado, esto sumado al alto crecimiento de la tecnología en la vida del ser humano [8] ha hecho que se tome el uso de nuevas tecnologías para el beneficio de la educación. En mención a lo expuesto en la actualidad se cuenta con tres tipos de software educativos: programas tutoriales, tutores inteligentes y simuladores que facilitan el conocimiento explícito [9] [10].

2.1.2. Entornos de simulación

Un simulador es un modelado que acopla diferentes técnicas [11], que facilita las prácticas de ensayo y error que hace posible la experimentación [12]; es uno de los métodos más recomendados considerado como un objeto de aprendizaje compuesto, que permite al alumno poder adquirir y reafirmar sus conocimientos mediante un entorno simulado compuesto de texto he imágenes, minorando la complejidad [3] [13].

2.1.3. Sistemas operativos

En el auge de la informática se cuenta con el desarrollo importante que tiene el hardware, el software y especialmente de los SO [14]. Los sistemas operativos es la comunicación que existe entre el hardware de un computador y la interfaz que logra la comunicación entre el lenguaje de máquina y el usuario [15].

2.1.4. Memoria RAM:

Memoria de Acceso Aleatorio (RAM), es aquella que usa el sistema operativo para procesar toda la información de nuestra computadora, tiene como función elemental tener los procesos para que el procesador los ejecute haciendo uso de una memoria virtual con técnicas de paginación o segmentación [16].

2.1.5. Administración de memoria

Dentro de los SO la administración de memoria, que no es más que la asignación de memoria a los diferentes programas que la requiera y liberar espacios de memoria inutilizables para dejarlos a disposición de otros programas, logrando un reparto transparente, eficiente y seguro [17].

2.1.6. Memoria virtual

La gestión de memoria por la necesidad de administrar ambientes de múltiples procesos, trata de obtener el máximo provecho de la memoria, tal es el caso que para realizarlo traslada la información de la CPU a la memoria principal [18], este hecho es también conocido como memoria virtual, puesto que no hace uso de la memoria física del procesador, sino de una memoria representativa. [19].

2.1.7. Paginación de memoria

Técnica de dividir la memoria en secciones físicas denominadas marcos de páginas, acompañada de la partición de los procesos en partes del mismo tamaño. Dicho de otra manera, las zonas en las que es dividido el proceso se denominan páginas y las zonas de la memoria se lo conocen como marco [16]. En algún momento un marco será llenado por una página de un proceso, lo que implica que estas dos siempre estén directamente relacionadas, esta estructura es conocida como tabla de páginas, es decir que estas tablas tendrán la identificación de la dirección física mediante el desplazamiento y el número de marco, en la ilustración 1 se muestra el esquema de traducción de la página.

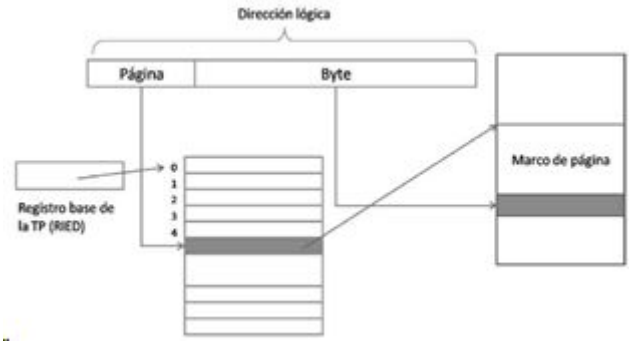


Ilustración 1: Esquema de traducción de la paginación. [16]

2.1.8. Fallos de página

Un fallo de página se presenta cuando se solicita ejecutar un proceso y este no se encuentra en memoria [5].

2.1.9. Algoritmo

Un algoritmo es un conjunto de pasos ordenados para conseguir una solución [20], un algoritmo computacional es complejo por su elaboración, sin embargo, entre los más complejos tenemos aquellos geométricos y aquellos que tienen que de una u otra forma relacionarse con la rama de las matemáticas [21] [22].

2.1.9.1.Algoritmo FIFO

FIFO que quiere decir first in first out, primero en entrar primero en salir [23] [24]. Este método de paginación trabaja conforme la llegada, cuando la memoria ya no tiene marcos disponibles comienza a funcionar este algoritmo sacando al primer proceso que haya entrado y colocando al final de la cola el que ingresa.

2.1.9.2.Algoritmo LRU

Este algoritmo consiste en asociar a cada bloque de datos la hora en la cual han sido utilizados por última vez. Luego, a la hora de realizarse el reemplazo de un bloque de datos en la Cache, se elige el bloque que hace más tiempo no ha sido utilizado.

2.2. Solución del problema

2.2.1. Materiales

Para la simulación del funcionamiento de un algoritmo, es necesario analizar detenidamente el comportamiento de los mismos y proporcionar al simulador los distintos parámetros que se consideraran en el sistema [6], para este caso se usó programación en lenguaje C# en el entorno de desarrollo de Microsoft Visual Studio. Y para realizar la lógica del algoritmo se ha utilizado todos los artículos que se encuentran referenciados en el desarrollo de este proyecto.

2.2.2. Diagrama de flujo

Para realizar la representación del proceso interno de los algoritmos de paginación se ha utilizado la herramienta yEd, ya que es una herramienta libre y de fácil entendimiento para hacer diagramas de flujo. Además, que ofrece descargar el archivo en diversos formatos una vez finalizado.

Algoritmo LRU è *(Ver anexo 1)*
Algoritmo FIFO è (Ver anexo 2)

2.2.3. Procedimiento

Siendo la memoria RAM una memoria de capacidad limitada donde se guardan los procesos que están en ejecutados en ese momento. Es decir que, si se está trabajando y se tiene Word, un navegador, Excel abierto, todos esos programas se guardan en la RAM porque la RAM 1es una memoria de acceso rápido a diferencia del disco duro que es algo mecánico que va a tardar más tiempo.

Cada programa se distribuye en páginas llamadas bloques o segmentos, estos son guardados en la memoria RAM. Para guardas estas páginas existen muchos métodos, para este caso se tomará el FIFO y el LRU. La función de estos algoritmos es de ver la forma de ordenar las páginas y tener la disponibilidad de las mejores e ir descartando las menos usadas.

Como se ha escrito en el transcurso de este documento, existen un sinnúmero de información plasmadas en libros, artículos y otros medios para adquirir este conocimiento de cómo funciona la memoria internamente, sin embargo, siguiendo las nuevas prácticas de aprendizaje, se crea un entorno de simulación mediante un programa, el cual su objetivo es dar un óptimo aprendizaje al alumnado [25].

En el programa se podrá observar en primera instancia la interfaz donde se podrá seleccionar la cantidad de números de referencia de páginas, que es la representación de la cantidad de programas que están actualmente en la memoria, estos estarán en un rango del 0 al 9 aleatoriamente; la cantidad de marcos de páginas, que es la representación de la capacidad de la memoria, estos estarán en un rango del 1 al 7 aleatoriamente. Adicional podemos observar un cuadro de texto donde se muestra las referencias de las páginas que se van a tomar, estas

se pueden dejar de forma aleatoria o sino se las puede ingresar manualmente deseleccionando el casillero donde dice aleatorio; También se puede observar dos botones que son los que determinaran que algoritmo se va a poner a funcionar con los elementos seleccionados anteriormente tales como FIFO y LRU. Por último, se muestra información acerca de la respuesta de los algoritmos, la cantidad de fallos que hubiera en memoria y el rendimiento que se tiene al usar el algoritmo.

2.2.3.1.Procedimiento de algoritmo FIFO

El algoritmo FIFO es, primer elemento que entra es el primero que sale, de tal manera que queda una estructura tipo de cola como la de un banco. Para este caso en el programa tenemos: 25 referencias de páginas, 5 marcos de páginas y se ha dejado seleccionada la casilla de aleatorio para que el programa dé las referencias aleatoriamente como se puede observar en la ilustración 2.

Ilustración 2: Datos para el algoritmo FIFO

Ø Empezando, ingresa la ref4, como la memoria está vacía se genera un fallo de página, dado que los fallos se generan cuando la CPU tiene que ingresar a memoria a buscar dicho proceso.

Ø Ingresa uno nuevo elemento que es la ref6, verifica si la memoria está vacía como ya tiene un elemento, entonces procede a revisar si existe el elemento, como no existe, verifica que

haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.

- Ø A continuación, ingresa la ref7, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø Luego ingresa ref7, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como si existe, no se genera un fallo de página.
- Ø A continuación, ingresa la ref1, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø Luego ingresa ref1, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como si existe, no se genera un fallo de página.
- Ø A continuación, ingresa la ref3, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø Luego ingresa ref6, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como si existe, no se genera un fallo de página.
- Ø A continuación, ingresa la ref2, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como no lo hay empieza a trabajar la esencia del algoritmo FIFO, primero en entrar primero en salir, se extrae el primer elemento que ingresó a memoria, en este caso ref4, se inserta el nuevo elemento al final de la cola y se genera un fallo de página.
- Ø El procedimiento es repetitivo en cada ingreso de una nueva referencia, siguiendo el diagrama de flujo *(Ver anexo 1)* se podrá obtener un buen aprendizaje del algoritmo.
- Ø Como se puede observar en pantalla del programa en la parte izquierda de la ventana se verá la cantidad total de fallos que se obtuvo mediante el algoritmo y el rendimiento del mismo expresado en porcentaje *como se observa en la ilustración 2.*

2.2.3.2. Procedimiento de Algoritmo LRU

El algoritmo LRU como su nombre lo indica hace la sustitución en la página menos utilizada recientemente. Para este caso en el programa tenemos: 25 referencias de páginas, 5 marcos de páginas y se ha dejado seleccionada la casilla de aleatorio para que el programa dé las referencias aleatoriamente como se puede observar en la ilustración 3.

Ilustración 3: Datos para el algoritmo LRU

- Ø Empezando, ingresa la ref3, como la memoria está vacía se genera un fallo de página, dado que los fallos se generan cuando la CPU tiene que ingresar a memoria a buscar dicho proceso.
- Ø Ingresa uno nuevo elemento que es la ref1, verifica si la memoria está vacía como ya tiene un elemento, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø A continuación, ingresa la ref7, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø Luego ingresa ref1, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como si existe, no se genera un fallo de página.
- Ø A continuación, ingresa la ref4, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya

espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.

- Ø A continuación, ingresa la ref8, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como si lo hay, entonces inserta el elemento a memoria y se genera un fallo de página.
- Ø A continuación, ingresa la ref5, verifica si la memoria está vacía como ya tiene elementos, entonces procede a revisar si existe el elemento, como no existe, verifica que haya espacio, como no lo hay empieza a trabajar la esencia del algoritmo LRU, se extrae el elemento menos usado recientemente en memoria, en este caso ref3, se inserta el nuevo elemento al final de la cola, se ordenan los elementos quedando como primer elemento el menos usado recientemente y al final se genera un fallo de página.
- Ø El procedimiento es repetitivo en cada ingreso de una nueva referencia, siguiendo el diagrama de flujo *(Ver anexo 2)* se podrá obtener un buen aprendizaje del algoritmo.
- Ø Como se puede observar en pantalla del programa en la parte derecha de la ventana se verá la cantidad total de fallos que se obtuvo mediante el algoritmo y el rendimiento del mismo expresado en porcentaje *como se observa en la ilustración 3.*

2.3. Resultados

2.3.1. Consideraciones teóricas

Considerando el auge de la tecnología es importante considerar las nuevas técnicas de aprendizaje [26], por medio de este proyecto queda comprobado que la técnica de creación de Entornos de Simulación no solo ayuda al desarrollo de las prácticas de fallo y error, sino también minora la complejidad del problema de estudio, tal es el caso que por medio del programa se simulación se puede observar que es lo que pasa realmente internamente en la memoria cuando se utiliza las técnicas de paginación de memoria con el algoritmo FIFO y LRU.

2.3.2. Consideraciones del diseño

En primera instancia se deja claro que la cantidad de los datos que se encuentran al inicio del programa se los puede cambiar según sea conveniente. El diseño del entorno de simulación es muy limpio, lo que permite que el estudiante logre su conocimiento de forma fácil y clara.

2.3.3. Solución de la propuesta

Por medio del proyecto expuesto en este documento se cumplió a cabalidad la propuesta de la parte práctica del examen complexivo donde se requería un programa con algoritmos de sustitución de páginas FIFO Y LRU, bajo los siguientes parámetros:

- Generar una cadena de referencia de páginas aleatoria en la que los números de página estén comprendidos entre 0 y 9.
- Aplicar la cadena aleatoria de referencias de página a cada algoritmo.
- Registrar el número de fallos de página provocados por cada algoritmo.
- Implementar los algoritmos de sustitución de modo que el número de marcos de página pueda variarse entre 1 y 7.

3. CONCLUSIONES

Es importante implementar las últimas tendencias tecnológicas respecto al aprendizaje, puesto que ayuda sobremanera a lograr comprender conceptos o procedimientos que podrían durar días y días, en solo cuestión de horas. Además, aplicándolo en los alumnos de menor edad ayudamos a que el aprendizaje sea creativo y didáctico.

Mediante la lectura de este proyecto se concluye con los siguientes conocimientos: la gestión de memoria es un proceso de asignación que se realiza en la ejecución de los programas, esta se libera cuando el programa haya sido cerrado. Un programa bien escrito utiliza la menor cantidad de memoria posible. La paginación logra un buen manejo entre la dirección lógica con la dirección física a la memoria.

El desarrollo de esta propuesta permitirá soportar los procesos de aprendizajes diferenciando las características de cada algoritmo de sustitución de las páginas de memoria.

REFERENCIAS BIBLIOGRÁFICAS

[1] K. Malave Polanco y J. L. Beauperthuy Taibo, «Android el sistema operativo de Google para dispositivos máviles,» *Negotium ,* vol. 7, nº 19, pp. 79-96, 2011.

[2] E. Puente FUentes, D. Hernández Balú, M. Hurtado Fedorovich y J. E. Torres Sánchez, «Nova como sistema operativo embebido para hardware cubano,» *Revista Cubana de Ciencias Informáticas,* vol. 6, nº 1, pp. 1-9, 2012.

[3] J. J. Dominguez y A. Estero , «Herramienta de apoyo a la docencia de Sistemas Operativos,» *Lenguajes y sistemas informáticos,* vol. 1, nº 16, pp. 147-156, 2013.

[4] D. J. Lanccheros C, A. Carrillo Ramos y J. L. Lara , «Modelo de adaptación en ambientes virtuales de aprendizaje para personas con discapacidad,» *Avances en Sistemas e Informática,* vol. 8, nº 2, pp. 17-30, 2011.

[5] A. E. Urrea Cardozo y R. Gómez, «Un modelo de memoria virtual compartida distribuida para Mapaná,» *Ingeniería y Competitividad,* vol. 7, nº 1, pp. 33 - 43, 2005.

[6] J. J. Dominguez, A. Estero y F. Veláazquez , «Simulador de algoritmos de sustitucion de paginas,» *Lenguajes y sistemas informáticos,* vol. 1, nº 16, pp. 189-213, 2013.

[7] F. J. Peñalvo García y L. V. Safont, «ASPECTOS PEDAGÓGICOS EN LA INFORMÁTICA EDUCATIVA,» *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información,* vol. 14, nº 2, pp. 371-375, 2013.

[8] E. Ávila Barrientos, «Formación de usuarios de la información mediante aplicaciones Web 2.0,» *Biblios,* nº 55, pp. 40-50, 2014.

[9] J. E. Ulises ROmán, «GESTIÓN DEL CONOCIMIENTO Y DE TECNOLOGÍAS DE INFORMACIÓN EN LA UNIVERSIDAD PERUANA,» *Investigación de Sistemas Informáticas,* pp. 93-101, 2004.

[10] J. J. Dominguez, A. Estero y M. Palomo, «Simulador de algoritmos de planificación de la CPU,» *Lenguajes y sistemas informáticos,* vol. 1, nº 16, pp. 115-126, 2013.

[11] R. Zúñiga Sáenz, «Operaciones: concepto, sistema, estrategia y simulación,» *Revista Latinoamericana de Administración,* nº 34, pp. 1-24, 2005.

[12] R. Zegarra, «SIMULACIÓN EN LA CAPACITACIÓN EN,» *Investigación de sistemas informáticos,* pp. 27-37, 2004.

[13] J. Montilva, C. M. Rojas y D. A. Orjuela, «RDOA-WS: repositorio distribuido de objetos de aprendizaje soportado con servicios web,» *Avances en Sistemas e Informática,* vol. 8, nº 2, pp. 183-189, 2011.

[14] J. L. Peña Catalá, W. Pérez Torres y Y. Blanco García, «El Sistema Operativo LINUX y el cambio necesario,» *Panorama Cuba y Salud,* vol. 1, nº 2, pp. 44-50, 2006.

[15] A. López Get, «La guerra de los sistemas operativos V1.0,» *Reflexiones,* vol. 88, nº 2, pp. 61-73, 2010.

[16] N. Y. Gélvez García , I. J. Ponce Fajardo y L. S. Rodríguez Castro, «Paginación de memoria en OS X,» *Redes de Ingeniería,* vol. 4, nº 2, pp. 34 - 39, 2013.

[17] C. E. Velázquez Rodríguez , «SMRP: Una aplicación informática para simular el reemplazo de páginas de memoria en los Sistemas Operativos,» *Ciencia & Futuro,* pp. 81 - 91, 2013.

[18] J. C. Gómez y N. D. Muñoz , «Hardware and software architecture for a Rover robot,» *Revista Avances en Sistemas e Informática,* vol. 8, nº 3, pp. 183-189, 2011.

[19] J. Navarro, «Simulación bajo memoria compartida de un sistema distribuido que simula memoria compartida,» *CACIC 97,* vol. 97, nº 2, pp. 31 - 45, 2015.

[20] L. Pro Concepción, N. Serna Palomino, A. Cortez Vásque, W. Contreras Flores, J. Díaz Muñante y G. Castro León , «Modelado y diseño con herramientas de complejidad, presentación de algoritmos y aplicaciones. Caso mercado laboral en la Carrera de Ingeniería de Sistemas -UNMSM,» *REVISTA DEbVESTIGACIÓl\: DE SISTE.\1ASE bFOR:.1ÁTICA,* vol. 10, nº 1, pp. 9-24, 2014.

[21] A. Ramírez López, D. Muñoz Negrón , A. Cruz Ramírez y Á. Morales Ramírez , «PROGRAMACIÓN DE ALGORITMOS COMPUTACIONALES PARA LA SIMULACIÓN DE ESTRUCTURAS DE GRANO,» *Avances en Ciencias e Ingeniería,,* vol. 3, nº 3, pp. 131-139, 2012.

[22] D. m. Manuel Tupia, «UN ALGORITMO VORAZ PARA RESOLVER EL PROBLEMA DE LA PROGRAMACIÓN DE TAREAS DEPENDIENTES EN MÁQUINAS DIFERENTES,» *Revista de Investigación y Sistemas Informáticos,* vol. 1, nº 1, pp. 9-18, 2004.

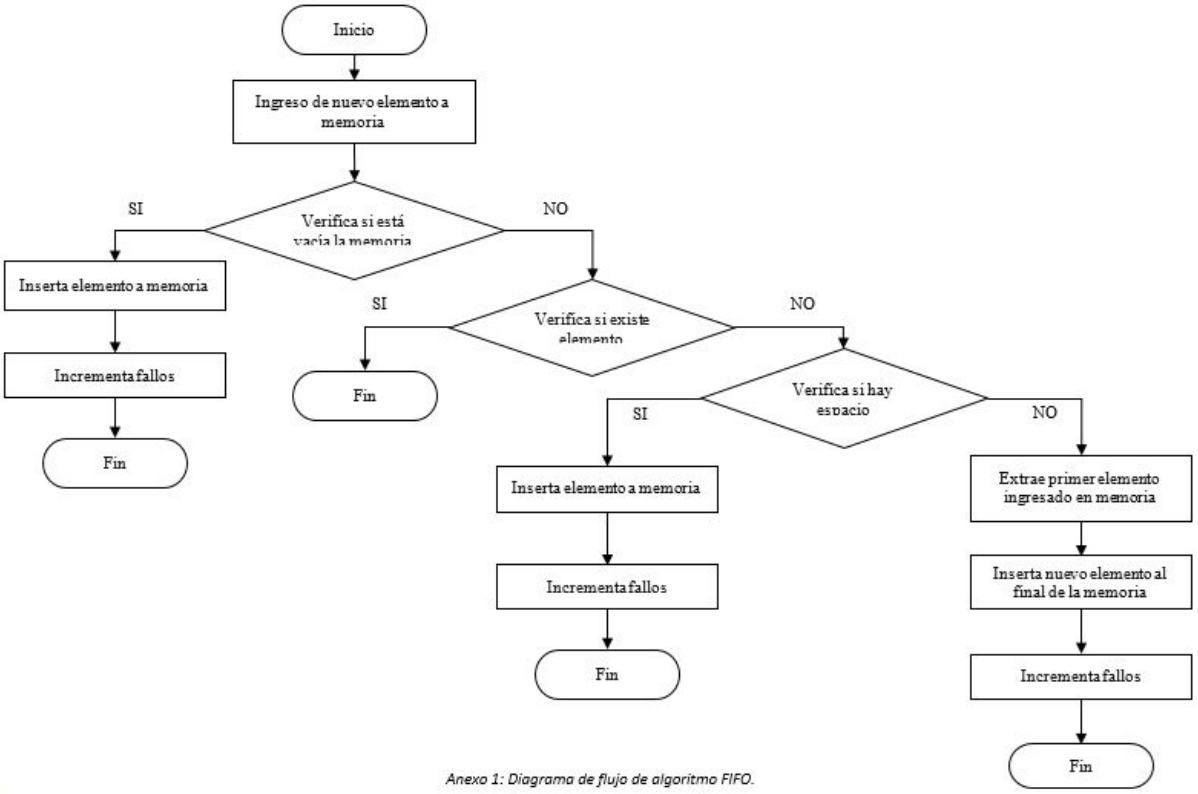
[23] J. E. Marquez Delgado, R. L. Ávila Rondón, M. Á. Gómez, E. González y C. R. Herrera Márquez, «Algoritmo genético aplicado a la programación en talleres de máquinado,» *Ingeniería Mecánica,* vol. 15, nº 3, pp. 201-212, 2012.

[24] N. Serna Palomino y L. Pró Concepción, «Un algoritmo eficiente y flexible para segmentación de imágenes de geles 2-DE,» *Revista de Investigación de Sistemas e Informática,* vol. 7, nº 2, pp. 35-41, 2010.

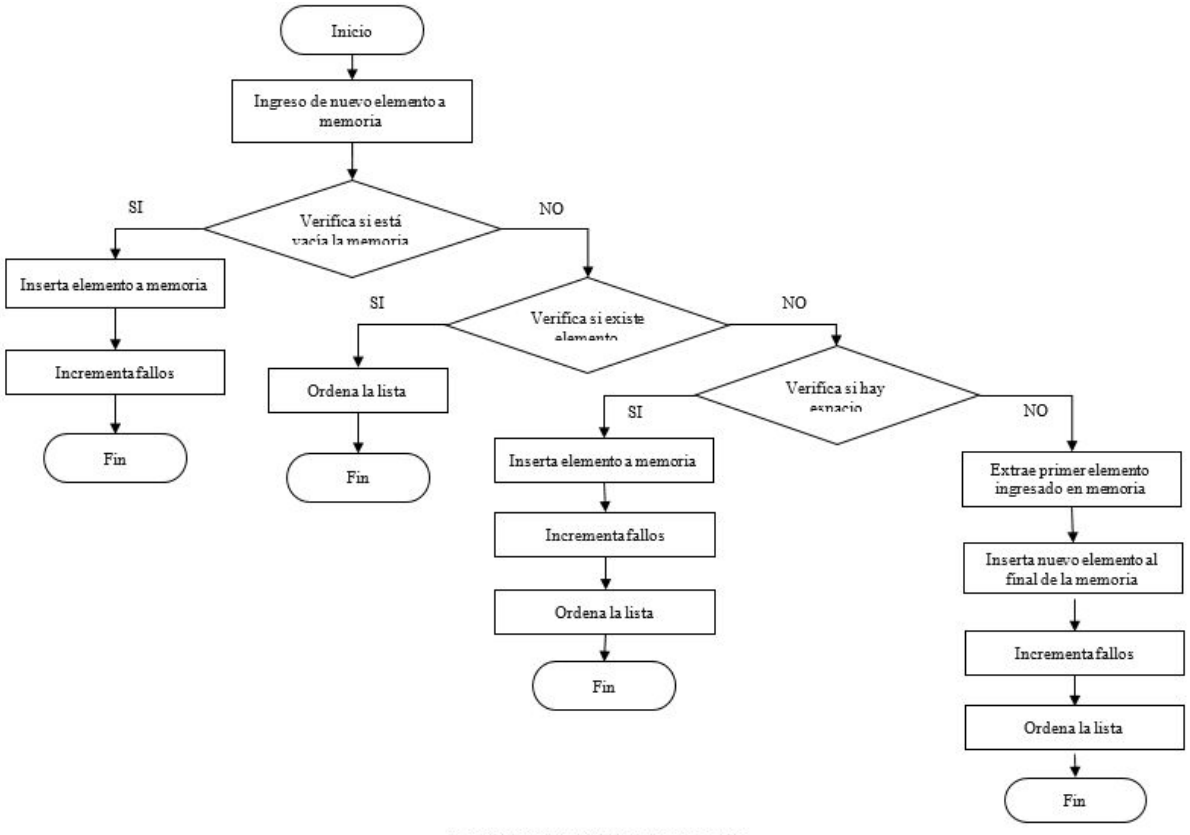
[25] L. Pró, M. E. Ruíz y C. Yanez, «DISEÑO E IMPLEMENTACIÓN DE UN MODELO DE APRENDIZAJE MEDIANTE ALGORITMOS GENÉTICOS,» *Investigación de Sistemas Informáticas,* pp. 21-29, 2005.

[26] J. R. Altamirano Yupanqu, «Riesgos TIC desde una Perspectiva Interdisciplinaria,» *REVISTA DE b,'ESTIGACIÓ~ DE SISTEMAS E I~FORMÁTICA,* vol. 10, nº 1, pp. 45-52, 2014.

ANEXOS



Anexo 1: Diagrama de flujo de algoritmo FIFO.



Anexo 2: Diagrama de flujo de algoritmo LRU