



**ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ
MANUEL FÉLIX LÓPEZ**

CARRERA INFORMÁTICA

**TRABAJO DE TITULACIÓN PREVIA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN INFORMÁTICA**

MODALIDAD: SISTEMATIZACIÓN DE EXPERIENCIAS

TEMA:

ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO

AUTORAS:

**CARMEN GEMA CEDEÑO COOL
DIANA RAQUEL LÓPEZ CEDEÑO**

TUTOR:

ING. LUIS CRISTÓBAL CEDEÑO VALAREZO, MGTR.

CALCETA, NOVIEMBRE 2017

DERECHOS DE AUTORÍA

Carmen Gema Cedeño Cool y Diana Raquel López Cedeño, declaran bajo juramento que el trabajo aquí descrito es de nuestra autoría, que no ha sido previamente presentado para ningún grado o calificación profesional, y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos los derechos de propiedad intelectual a la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, según lo establecido por la Ley de Propiedad Intelectual y su reglamento.

.....
CARMEN G. CEDEÑO COOL

.....
DIANA R. LÓPEZ CEDEÑO

CERTIFICACIÓN DEL TUTOR

Luis Cristóbal Cedeño Valarezo certifica haber tutelado el trabajo de titulación ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO, que ha sido desarrollado por Carmen Gema Cedeño Cool y Diana Raquel López Cedeño, previa la obtención del título de Ingeniero en Informática, de acuerdo al REGLAMENTO PARA LA ELABORACIÓN DE TRABAJO DE TITULACIÓN DE TERCER NIVEL de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

.....
ING. LUIS C. CEDEÑO VALAREZO, MGTR.

APROBACIÓN DEL TRIBUNAL

Los suscritos integrantes del tribunal correspondiente, declaran que han APROBADO el trabajo de titulación ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO, que ha sido propuesto, desarrollado y sustentado por Carmen Gema Cedeño Cool y Diana Raquel López Cedeño, previa la obtención del título de Ingeniero en Informática, de acuerdo al REGLAMENTO PARA LA ELABORACIÓN DE TRABAJO DE TITULACIÓN DE TERCER NIVEL de la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López.

.....

ING. ORLANDO AYALA PULLAS, MGTR.

MIEMBRO

.....

ING. LIGIA E. ZAMBRANO SOLORZANO, MGTR.

MIEMBRO

.....

ING. DANIEL A. MERA MARTÍNEZ, MGTR.

PRESIDENTE

AGRADECIMIENTO

A la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López que nos dio la oportunidad de una educación superior de calidad y en la cual hemos forjado nuestros conocimientos profesionales día a día;

A nuestro tutor Ing. Luis Cedeño Valarezo por brindarnos su valiosa colaboración, por su asesoría y apoyo en la realización de este trabajo de titulación, y

A los docentes, que por sus conocimientos impartidos y motivación, con el pasar de los años se convirtieron en un ejemplo a seguir.

LAS AUTORAS

DEDICATORIA

A Dios por brindarnos salud y habernos permitido llegar a este momento tan especial.

A nuestros padres por habernos brindado su apoyo moral y económico de manera incondicional para poder lograr nuestros objetivos.

A nuestras hermanas que siempre han estado brindándonos apoyo y dándonos ánimos para no decaer.

A nuestras familias por apoyarnos en los buenos y malos momentos, por compartir sus experiencias y orientarnos hacia sendas correctas, llenas de valores y ética.

LAS AUTORAS

CONTENIDO GENERAL

CARÁTULA.....	i
DERECHOS DE AUTORÍA.....	ii
CERTIFICACIÓN DEL TUTOR	iii
APROBACIÓN DEL TRIBUNAL.....	iv
AGRADECIMIENTO	v
DEDICATORIA.....	vi
CONTENIDO GENERAL.....	vii
RESUMEN	xii
PALABRAS CLAVE	xii
ABSTRAC.....	xiii
KEY WORDS.....	xiii
CAPÍTULO I. ANTECEDENTES.....	1
DESCRIPCIÓN DE LA INSTITUCIÓN.....	1
DESCRIPCIÓN DE LA INTERVENCIÓN	3
1.1. OBJETIVOS.....	5
1.1.1. OBJETIVO GENERAL.....	5
1.1.2. OBJETIVOS ESPECÍFICOS.....	5
CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN	6
2. 1. DEFINIR LOS ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO	7
2. 2. ANALIZAR LOS ALGORITMOS.....	9
2. 3. ESTABLECER LA EFICIENCIA Y EFICACIA DE LOS MISMOS.....	12
CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA.....	15
3.1. DEFINIR LOS ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO.....	15
3.2. ANALIZAR LOS ALGORITMOS	15

3.3. ESTABLECER LA EFICIENCIA Y EFICACIA DE LOS MISMOS	34
3.3.1. CANTIDAD DE ACIERTOS	37
3.3.1.1. IDE	37
3.3.1.2. ALGORITMOS.....	38
3.3.2. ALMACENAMIENTO (RAM).....	40
3.3.2.1. IDE	40
3.3.2.2. ALGORITMOS.....	43
3.3.3. PROCESAMIENTO	44
3.3.3.1. IDE	44
3.3.3.2. ALGORITMOS.....	45
3.3.4. TIEMPO DE RESPUESTA	46
3.3.4.1. IDE	46
3.3.4.2. ALGORITMOS.....	47
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	51
4.1. CONCLUSIONES.....	51
4.2. RECOMENDACIONES.....	51
BIBLIOGRAFÍA.....	52
ANEXOS	55

CONTENIDO DE CUADROS

Cuadros

CUADRO 2. 1. COMPARACIÓN DE ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO.	8
CUADRO 2. 2. FORMATO DE FICHA DE OBSERVACIÓN.....	10
CUADRO 2. 3. FORMATO DE LOS INDICADORES DE EFICACIA Y EFICIENCIA.	11

CUADRO 2. 4. FORMATO PARA EL REGISTRO DE RESULTADOS DE LA COMPILACIÓN DE ALGORITMOS EN DIFERENTES COMPUTADORAS	12
CUADRO 2. 5. FORMATO DE CUADRO COMPARATIVO DE LOS ALGORITMOS.	13
CUADRO 3. 1. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 1.....	16
CUADRO 3. 2. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 1.....	17
CUADRO 3. 3. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 1.....	18
CUADRO 3. 4. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 2.....	19
CUADRO 3. 5. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 2.....	20
CUADRO 3. 6. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 2.....	21
CUADRO 3. 7. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 3.....	22
CUADRO 3. 8. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 2.....	23
CUADRO 3. 9. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 3.....	24
CUADRO 3. 10. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 4.....	25
CUADRO 3. 11. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 4.....	26
CUADRO 3. 12. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 4.....	27
CUADRO 3. 13. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 5.....	28
CUADRO 3. 14. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 5.....	29
CUADRO 3. 15. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 5.....	30

CUADRO 3. 16. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN C-FREE - COMPUTADORA 6.....	31
CUADRO 3. 17. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN DEV-C++ - COMPUTADORA 6.....	32
CUADRO 3. 18. RESULTADOS DE LOS ALGORITMOS COMPILADOS EN CODEBLOCKS - COMPUTADORA 6.....	33
CUADRO 3. 19. RESULTADO DEL ANÁLISIS DEL ALGORITMO MAPREDUCE EN GOOGLE CLOUD PLATFORM.....	34
CUADRO 3. 20. CUADRO COMPARATIVO DE LOS ALGORITMOS E IDE.	35
CUADRO 3. 21. CUADRO COMPARATIVO DE IDE Y ALGORITMOS.....	36
CUADRO 3. 22. CUADRO DE RESULTADOS DE MAPREDUCE.....	37
CUADRO 3. 23. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE IDE.....	37
CUADRO 3. 24. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE IDE.....	38
CUADRO 3. 25. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE ALGORITMOS.....	38
CUADRO 3. 26. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE ALGORITMOS.....	39
CUADRO 3. 27. MEDIA ARITMÉTICA DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE ALGORITMOS.	39
CUADRO 3. 28. PRUEBA TUKEY INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE ALGORITMOS.....	40
CUADRO 3. 29. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR ALMACENAMIENTO CON LA VARIABLE IDE.	41
CUADRO 3. 30. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR ALMACENAMIENTO CON LA VARIABLE IDE.	41
CUADRO 3. 31. MEDIA ARITMÉTICA DE INDICADOR ALMACENAMIENTO CON LA VARIABLE IDE.....	42
CUADRO 3. 32. PRUEBA TUKEY INDICADOR ALMACENAMIENTO CON LA VARIABLE IDE.	42
CUADRO 3. 33. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR ALMACENAMIENTO CON LA VARIABLE ALGORITMOS.....	43

CUADRO 3. 34. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR ALMACENAMIENTO CON LA VARIABLE ALGORITMOS.....	44
CUADRO 3. 35. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR PROCESAMIENTO CON LA VARIABLE IDE.	44
CUADRO 3. 36. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR PROCESAMIENTO CON LA VARIABLE IDE.....	45
CUADRO 3. 37. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR PROCESAMIENTO CON LA VARIABLE ALGORITMOS.	45
CUADRO 3. 38. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR PROCESAMIENTO CON LA VARIABLE ALGORITMOS.....	46
CUADRO 3. 39. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE IDE.....	47
CUADRO 3. 40. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE IDE.....	47
CUADRO 3. 41. ANÁLISIS DE VARIANZA DE DOS FACTORES DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE ALGORITMOS.....	48
CUADRO 3. 42. ANÁLISIS DE VARIANZA DE UN FACTOR DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE ALGORITMOS.....	48
CUADRO 3. 43. MEDIA ARITMÉTICA DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE ALGORITMOS.	49
CUADRO 3. 44. PRUEBA TUKEY INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE ALGORITMOS.....	49

GRÁFICOS

GRÁFICO 3. 1. REPRESENTACIÓN DE MEDIAS DE PRUEBA TUKEY DE INDICADOR CANTIDAD DE ACIERTOS CON LA VARIABLE ALGORITMOS.....	40
GRÁFICO 3. 2. REPRESENTACIÓN DE MEDIAS DE PRUEBA TUKEY DE INDICADOR ALMACENAMIENTO CON LA VARIABLE IDE.	43
GRÁFICO 3. 3. REPRESENTACIÓN DE MEDIAS DE PRUEBA TUKEY DE INDICADOR TIEMPO DE RESPUESTA CON LA VARIABLE ALGORITMOS.	50

RESUMEN

El presente trabajo de titulación tuvo como propósito determinar la eficiencia de los algoritmos de búsqueda de patrones de textos: Fuerza Bruta, Boyer-Moore-Horspool, Knuth- Morris-Pratt y MapReduce, para establecer cuál o cuáles presentaban un mejor comportamiento. Para cumplir con el desarrollo del mismo fue necesario emplear los métodos: bibliográfico, analítico y diseño experimental con análisis de varianza con varios factores. A través del método bibliográfico se pudo definir los algoritmos de búsqueda de patrones de texto, posteriormente se realizó el análisis de los algoritmos con el método analítico donde se emplearon los siguientes indicadores: cantidad de aciertos, recursos para almacenar y procesar (memoria RAM) y tiempo de respuesta, luego se compilaron los algoritmos en seis computadoras diferentes con las IDE (Integrated Development Environment, por sus siglas en inglés): C-Free, Dev-C++ y CodeBlocks. Finalmente, se estableció la eficiencia y eficacia de los algoritmos por medio de diseño experimental con análisis de varianza con varios factores. El procedimiento antes descrito permitió establecer que en los indicadores: cantidad de aciertos y tiempo de respuesta el algoritmo Fuerza Bruta es más eficiente y eficaz, mientras que la IDE CodeBlocks presentó mejor comportamiento con respecto a los algoritmos.

PALABRAS CLAVE

Algoritmos de búsqueda, patrones en textos, fuerza bruta.

ABSTRAC

The purpose of this study was to determine the efficiency of the algorithms for searching text patterns: Brute Force, Boyer-Moore-Horspool, Knuth-Morris-Pratt and MapReduce, in order to establish which one or which showed the best behavior. To comply with its development it was necessary to use the methods: bibliographic, analytical and experimental design with analysis of variance with several factors. Through the bibliographic method it was possible to define algorithms to search for text patterns. Afterwards, the analysis of the algorithms was performed with the analytical method, where the following indicators were used: number of hits, resources to store and process (RAM) and the response time, then the algorithms were compiled on six different computers with the IDE (Integrated Development Environment): C-Free, Dev-C ++ and CodeBlocks. Finally, the efficiency and effectiveness of the algorithms were established through experimental design with analysis of variance with several factors. The procedure described above allowed us to establish that in the indicators: on number of hits and response time the Brute Force algorithm is more efficient, while IDE CodeBlocks presented better behavior with respect to the algorithms.

KEY WORDS

Search algorithms, patterns in texts, brute force.

CAPÍTULO I. ANTECEDENTES

DESCRIPCIÓN DE LA INSTITUCIÓN

La Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López ESPAM MFL (2015a) nació con la necesidad de crear un centro de estudios superiores en las áreas agrícolas y pecuarias, de manera que la población estudiantil, con dificultad para trasladarse a universidades fuera de la zona, pudiera alcanzar un título académico, a fin de servir más tarde, no solo al cantón, sino a toda la región.

Según la historia de la ESPAM MFL (2015a) que reposa en la página web de la institución, las gestiones empezaron en el Congreso Nacional y luego en otras instancias desde 1995. El 29 de abril de 1996 se crea el INSTITUTO TECNOLÓGICO SUPERIOR AGROPECUARIO DE MANABÍ, ITSAM, mediante Ley N°. 116, publicada en el R.O. N°. 935. Y es así que en el 1999, el Congreso Nacional expidió la Ley Reformatoria que transformaba el Instituto Tecnológico Superior Agropecuario de Manabí, ITSAM, en ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE MANABÍ, ESPAM, cuya Ley 99-25 fue publicada en el R.O. el 30 de abril de 1999.

La reseña histórica señala que la ESPAM MFL (2015a) nace como persona jurídica de derecho público, autónoma, que se rige por la Constitución Política del Estado, Ley de Educación Superior, su Estatuto Orgánico y Reglamentos para preparar a la juventud ecuatoriana y convertirla en profesionales, conforme lo exigen los recursos naturales de su entorno. Iniciando sus labores con las Carreras de Agroindustrias, Medio Ambiente, Agrícola y Pecuaria. Luego mediante un estudio de mercado, se crea la Carrera de Informática emprendiendo así, un riguroso programa de fortalecimiento académico, con el fin de formar profesionales que ejecuten proyectos sustentables, que generen fuentes de trabajo.

De acuerdo a los antecedentes de la ESPAM MFL (2015a) ante la demanda de nuevas carreras, los directivos de la ESPAM, no escatimaron esfuerzos para incrementar otras, de tipo empresarial. Es así como desde el año 2003 funcionan nuevos programas: Administración Pública y Administración de Empresas, y la Carrera de Informática. A partir del año 2007 y, producto de un estudio, los estudiantes tienen una nueva opción: Ingeniería en Turismo.

Según el Modelo Educativo de la ESPAM MFL (2017) indica que la misión de la institución es: Formar profesionales pertinentes con compromiso ético y social, garantizándolo desde la calidad de las funciones sustantivas. Y como visión: Ser un centro de referencia en la capacitación de profesionales que contribuyan al desarrollo agropecuario y regional.

La Carrera de Computación ESPAM MFL (2015b) actualmente, se ha vuelto imprescindible en casi todos los campos en los que se desenvuelven las personas, sobre todo en ámbitos estratégicos y de interés general, responde a la necesidad de generar conocimientos que contribuyan a la transformación de la matriz productiva haciendo uso de la industria tecnológica de servicio; software, hardware y servicios informáticos. La misión de la carrera es: Formación de profesionales íntegros que conjuguen ciencia, tecnología y valores en su accionar, comprometidos con la comunidad en el manejo adecuado de herramientas computacionales de última generación. Y como visión: Ser referentes en la formación de profesionales de prestigio en el desarrollo de aplicaciones informática y soluciones de hardware.

DESCRIPCIÓN DE LA INTERVENCIÓN

Según Joyanes (2013) el mundo digital es considerado el repositorio de información más grande que existe, debido a que es en dicho mundo donde se genera gran cantidad de información gracias a las herramientas de comunicación, sistemas de procesamiento de información, medios virtuales, y otros. Así Rodríguez y Simón (2013), indican que en la web se encuentra alojada gran cantidad de información en formato texto.

Como indica Franco (2013) los algoritmos son procedimientos que se desarrollan con el fin de resolver problemas, los pasos deben ser concretos y su longitud finita, es por esta razón que se dice que cuando se programa, en realidad se está construyendo una instancia de un algoritmo. Así Bauer y Vega (2015) consideran que los algoritmos se construyen con el fin de minimizar las operaciones de entrada y salida.

Los datos se presentan de diferentes maneras y el texto es la principal forma para intercambiar información (Sánchez, 2012). Mientras Arcila *et al.*, (2016) indica que cuando los datos textuales han sido recogidos en gran cantidad es necesario emplear métodos computacionales que permitan realizar análisis de datos y obtener información relevante.

Ramírez (2014) establece que existen varios algoritmos de búsqueda de patrones de texto, pero no todos aprovechan la abundancia de cantidades de datos que nos rodean.

Lin y Dyer (2010) consideran que es importante conocer cuáles son los algoritmos de búsqueda de patrones de texto, porque los más eficaces son los que emplean de mejor manera la información. Por dicho motivo se realizó este trabajo de titulación que consistió en realizar una investigación exploratoria de los algoritmos de búsqueda de patrones de texto para determinar la eficacia y eficiencia de cada uno de estos, mediante el análisis de los algoritmos; posteriormente se determinaron indicadores que permitieron estudiar su

comportamiento frente a los volúmenes de datos para establecer la eficiencia de los mismos.

En el ámbito social permite brindar información sobre los algoritmos de búsqueda de patrones de texto más eficientes y eficaces. En lo económico evita la pérdida de información valiosa que es desechada por no estar al tanto de cómo enfrentar el volumen de datos en formato de texto.

El presente trabajo de titulación es muy favorable ya que las autoras no han encontrado una investigación de este tipo donde se haya realizado una comparación entre algoritmos de búsqueda de patrones de texto que indique la eficiencia y eficacia de los mismos. Cabe recalcar que este trabajo de titulación forma parte de un proyecto de investigación que tiene como título: Procesamiento de textos con gran cantidad de datos. Este trabajo se realizó en la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, el cual, beneficia directamente a la Unidad de Producción de Software, que es la unidad de docencia, investigación y vinculación de la Carrera de Computación que se encarga de brindar asesoría a los proyectos que se realizan en la misma, la cual, tiene a su disposición esta información para cuando necesite emplear algoritmos de búsqueda de patrones de texto.

1.1. OBJETIVOS

1.1.1. OBJETIVO GENERAL

Determinar la eficiencia de los algoritmos de búsqueda de patrones en textos, para establecer cuál o cuáles presentan un mejor comportamiento.

1.1.2. OBJETIVOS ESPECÍFICOS

- Definir los algoritmos de búsqueda de patrones de texto
- Analizar los algoritmos
- Establecer la eficiencia y eficacia de los mismos.

CAPÍTULO II. DESARROLLO METODOLÓGICO DE LA INTERVENCIÓN

Para la ejecución de este trabajo de titulación se empleó el método bibliográfico que según Porre (2013), es una excelente introducción a las investigaciones debido a que constituye la primera etapa de toda indagación, proporcionando conocimientos de trabajos existentes sobre el tema a resolver. Por otra parte Botero (2015) indica que el método bibliográfico consiste en la captación de datos con el fin de realizar un análisis crítico que construya procesos coherentes y abstracción compleja de los fenómenos, para así valorar o apreciar nuevas circunstancias

También se empleó el método analítico que según Maya (2014) es útil para llevar a cabo trabajos de investigación documental, en contraste, según Abreu (2014) es un método fundamental para toda investigación e imprescindible para realizar operaciones teóricas como son la conceptualización y la clasificación, mientras que Lopera *et al.*, (2010) citado por Pérez y Lopera (2016), establece que el método analítico es un procedimiento para llegar a la comprensión mediante la descomposición de un fenómeno en sus elementos constitutivos.

Sánchez (2009) citado por Garza (2013) el diseño de experimentos son métodos empleados en la manipulación de procesos, cuyo propósito es obtener información sobre cómo mejorarlo, mientras que Manterola y Otzen (2015) establece que el diseño experimental se caracteriza por la valoración del efecto de una o más intervenciones. Dentro del diseño experimental también se aplicó: análisis de varianza de dos factores con varias muestras y análisis de varianza con un factor. Según Rivera (2013) el análisis de varianza de dos factores con varias muestras, indica si los valores de una variable dependiente dependen de los niveles de los dos factores o si depende de la interacción de ambos. Rubio y Berlanga (2012) señalan que el análisis de varianza de un factor, también conocido como ANOVA de un factor, es el análisis que permite conocer si hay diferencia de medias en los grupos y explorar en que grupos

surgen dichas diferencias, si del análisis de varianza se rechaza la hipótesis nula, se debe continuar con la realización de los contrastes posteriores "post hoc", de este tipo de pruebas la más utilizada es la prueba Tukey, que según Navarro *et al.*, (2017) se usa cuando existen más de dos grupos a comparar, incrementando el número de pares de comparación y el nivel de significancia aumentará a más de 0,05. Para realizar la prueba Tukey se utilizó SPSS, del cual Torrado y Berlaga (2013) indican que es un paquete para el desarrollo de modelos estadísticos, estimaciones e interpretaciones de funciones discriminantes y validación de resultados.

2. 1. DEFINIR LOS ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO

El primer objetivo planteado por las autoras consistió en buscar información sobre los algoritmos de búsqueda de patrones de texto donde se empleó el método bibliográfico. A continuación se presenta el cuadro comparativo de los algoritmos de búsqueda de patrones de texto investigados:

Cuadro 2. 1. Comparación de algoritmos de búsqueda de patrones de texto.

Comparación de algoritmos de búsqueda de patrones de texto			
Nº	Algoritmo	Autor	Concepto
1	Fuerza Bruta	Moreno, E y Rubilar, F. 2015	Consiste en alinear el patrón a buscar con el texto e ir comparando de izquierda a derecha si los caracteres coinciden, si no lo hacen se desplaza el patrón una posición a la derecha.
2	Aho-Corasick	Cantos, P. 2012	Consiste en la búsqueda de un texto a través de una máquina de estados finitos, previamente creada analizando el conjunto de patrones a buscar. La máquina de estados es de tipo DFA, de esta forma se evita el backtracking presente en los NFA.
3	Boyer-Moore Hospool	Moreno, E y Rubilar, F. 2015	Se compara cada carácter de derecha a izquierda al contrario de otros algoritmos, en caso de fallar se hace coincidir el último carácter del texto que está alineado con el patrón del mismo. En caso de no existir el patrón se desplaza a la derecha
4	Shift-Or	Ramírez, K. 2014	Trabaja con un conjunto de caracteres y utiliza una máquina de estados, realiza la búsqueda iniciando con el carácter más la derecha
5	Knuth-Morris-Pratt	Torres, M; Páez, P; Martínez, C; Rodríguez, J. 2012	Realiza una comparación simple de izquierda a derecha y cuando no encuentra ocurrencia se desplaza una posición a la derecha para volver a realizar la comparación
6	MapReduce	Reutter, J. 2015.	Mapreduce es un modelo de programación orientado al procesamiento de grandes volúmenes de datos, desarrollados para sacar ventaja del poder de procesamiento paralelo que ofrecen los clúster de computadores
7	Rabin-Karp	Cantos, P. 2012	Este algoritmo usa un filtro Bloom para encontrar cualquiera de los patrones de un diccionario. Hay que tener en cuenta lo explicado en el apartado de filtros Bloom, y es que para trabajar con funciones hash es necesario que todos los patrones tengan la misma longitud, o al menos una longitud mínima
8	Boyer-Moore Sunday	Torres, M; Páez, P; Martínez, C; Rodríguez, J. 2012	El algoritmo BMS desliza el patrón basado en el símbolo del texto que corresponde a la posición del último carácter del patrón. Este siempre se desliza al menos una posición si se encuentra una discrepancia con el texto.

Considerando los criterios antes mencionados, se definió mediante una lista que los algoritmos a ser analizados son los siguientes: Fuerza bruta, Boyer-Moore Hospool, Knuth-Morris-Pratt, MapReduce; ya que de acuerdo al cuadro comparativo 2.1., estos realizan una búsqueda más simple y son los más utilizados. A continuación se presenta un pequeño concepto de estos algoritmos.

- **Fuerza bruta:** Consiste en alinear el patrón a buscar con el texto e ir comparando de izquierda a derecha si los caracteres coinciden, si no lo hacen se desplaza el patrón una posición a la derecha (Moreno y Rubilar, 2015).
- **Boyer-Moore-Horspool:** Según Moreno y Rubilar (2015) este algoritmo compara cada carácter de derecha a izquierda al contrario de otros algoritmos, en caso de fallar se hace coincidir el último carácter del texto que está alineado con el patrón del mismo. En caso de no existir el patrón se desplaza a la derecha.
- **Knuth-Morris-Pratt:** Realiza una comparación simple de izquierda a derecha y cuando no encuentra ocurrencia se desplaza una posición a la derecha para volver a realizar la comparación (Torres *et al.*, 2012).
- **MapReduce:** Reutter (2015) establece que MapReduce es un modelo de programación orientado al procesamiento de grandes volúmenes de datos, desarrollado específicamente para sacar ventaja del poder de procesamiento paralelo que ofrecen los clústers de computadores, mientras que Rosa y Rivera (2016) indica que MapReduce se encarga de gestionar mecanismos para tareas distribuidas en diferentes nodos del clúster Hadoop, esto permite realizar tareas de forma más sencilla y eficiente.

Una vez identificados los algoritmos a analizar se investigaron los códigos fuentes de los mismos.

2. 2. ANALIZAR LOS ALGORITMOS

El segundo objetivo estuvo conformado por dos actividades, en las que se empleó el método analítico. La primera actividad fue estudiar el funcionamiento de los algoritmos para lo cual se realizó una ficha de observación que contiene lo siguiente: nombre de algoritmo, referencia bibliográfica, indicadores de eficacia e indicadores de eficiencia.

Cuadro 2. 2. Formato de Ficha de observación.

FICHA DE OBSERVACIÓN PARA EL ANÁLISIS DE LOS ALGORITMOS						
Objetivo: Analizar los algoritmos de búsqueda de patrones de texto				Fecha:		
Algoritmos		Referencia bibliográfica		Indicador de eficacia	Indicadores de eficiencia	
					Cantidad de recursos que utiliza	
Nombre	Autor	Link		Cantidad de aciertos	Para almacenar (RAM)	Para procesar (RAM)
						Tiempo de respuesta
Fuerza Bruta						
Boyer-Moore-Horspool						
Knuth-Morris-Pratt						
MapReduce						

La segunda actividad estuvo enfocada en determinar los indicadores de eficiencia y eficacia de los algoritmos, para ello se desarrollaron los indicadores correspondientes. El indicador de eficacia, abarca la cantidad de aciertos que logra el algoritmo mientras el indicador de eficiencia incluye la cantidad de recursos (RAM) que emplean los algoritmos para almacenar y procesar, además del tiempo de respuesta.

Cuadro 2. 3. Formato de los indicadores de eficacia y eficiencia.

LISTADO DE INDICADORES DE EFICACIA Y EFICIENCIA DE LOS ALGORITMOS					
No.	NOMBRE DEL INDICADOR	TIPO DE INDICADOR	PARA QUE SIRVE EL INDICADOR	RESULTADO	RESPONSABLE
1	Cantidad de aciertos	EFICACIA	Cantidad de coincidencias encontradas en un texto		
2	Recursos para almacenar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para almacenar		
3	Recursos para procesar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para el procesamiento.		
4	Tiempo de respuesta	EFICIENCIA	Medir el tiempo que emplea el algoritmo en mostrar el resultado		

Para analizar los algoritmos: Fuerza bruta, Boyer-Moore Hospool, Knuth-Morris-Pratt se compilaron en lenguaje C++ empleando los IDE C-Free, Dev-C++, se iba a emplear Borland C++ pero surgieron inconvenientes con using namespace std; el cual sirve para agregar nuevas palabras reservadas y por ser Borland un IDE antiguo no permitió la compilación de los algoritmos, debido a esto se optó por usar CodeBlocks que es un IDE que incluye el compilador MinGW y es más moderno. Cabe recalcar que para el análisis de los algoritmos: Fuerza Bruta, Boyer-Moore Hospool y Knuth-Morris-Pratt, se utilizaron 6 computadoras con Sistema Operativo Windows las cuales poseen diferentes características, con esto se determinó el desempeño de los algoritmos.

Para la ejecución se usó un párrafo de texto de 497 palabras, posteriormente, se ingresó el patrón a buscar y luego la consola indicó los aciertos encontrados y el tiempo de respuesta. Para ver la memoria empleada por cada algoritmo al momento de almacenar y procesar se accedió al Administrador de Tareas del

Sistema Operativo de Windows. Sin embargo, el algoritmo MapReduce fue analizado una sola vez mediante la herramienta Google Cloud Platform, para habilitar esta cuenta fue necesario hacer uso de la tarjeta de crédito del Tutor, cabe indicar que dicha cuenta tiene 1 año de prueba. Para analizar este algoritmo se empleó el servicio BigQuery que es usado en el procesamiento de grandes volúmenes de datos, en este caso se usó una de las bases de datos que posee dicho servicio y por medio de una consulta BigQuery se procedió a buscar el patrón de texto, cuando se ejecutó la consulta se obtuvieron las coincidencias, el tiempo de respuesta así mismo el recurso de procesamiento que el algoritmo emplea.

Cuadro 2. 4. Formato para el registro de resultados de la compilación de algoritmos en diferentes computadoras

Computadora:	- Sistema Operativo:			
	- Procesador:			
	- Memoria instalada (RAM):			
	- Tipo de Sistema:			
IDE				
	Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM)	Para procesar (RAM)	Tiempo de respuesta
Fuerza Bruta				
Boyer-Moore- Horspool				
Knuth-Morris-Pratt				

2. 3. ESTABLECER LA EFICIENCIA Y EFICACIA DE LOS MISMOS

El último objetivo estuvo comprendido por siete actividades, de las cuales las dos primeras corresponden al desarrollo de la investigación y las cinco restantes pertenecen a la redacción del informe. La primera actividad de este objetivo consistió en la comparación de los algoritmos, para lo cual se realizó un cuadro comparativo con los indicadores mencionados en el apartado 2.2, en dicho cuadro se describen los algoritmos y los diferentes IDE empleados,

también se agruparon los resultados obtenidos en la compilación de los algoritmos que se efectuó en las seis máquinas.

Cuadro 2. 5. Formato de cuadro comparativo de los algoritmos.

CUADRO COMPARATIVO DE LOS ALGORITMOS												
	C-Free			Dev-C++			CodeBlocks					
Algoritmo	Cantidad de aciertos	Almacenar (RAM)	Procesar (RAM)	Tiempo de respuesta	Cantidad de aciertos	Almacenar (RAM)	Procesar (RAM)	Tiempo de respuesta	Cantidad de aciertos	Almacenar (RAM)	Procesar (RAM)	Tiempo de respuesta
Fuerza Bruta												
Boyer-Moore												
Horspool												
Knuth-Morris-Pratt												

La segunda actividad de este objetivo consistió en establecer la eficiencia y eficacia de los algoritmos, para lo cual se aplicó el diseño experimental con análisis de varianza con varios factores.

Para realizar el análisis se utilizó la herramienta Análisis de Datos de Microsoft Excel, en donde se utilizó la función: Análisis de varianza de dos factores con varias muestras por grupo, dicha función requiere el ingreso del rango de entrada y el número de filas por muestras. En el rango de entrada se ingresaron los datos a analizar y en la cantidad de filas por muestras se ingresó el número seis, debido a que dichas filas representan la cantidad de máquinas que se utilizaron para realizar la compilación de los algoritmos. El procedimiento antes mencionado se realizó para obtener un análisis por algoritmo y por IDE y posteriormente se efectuó el análisis de varianza de un factor, este análisis se realizó con el fin de analizar los resultados por separado, para ello se plantearon dos hipótesis para comprobar si los algoritmos e IDE producen resultados iguales y la segunda hipótesis formulada fue para comprobar si existe por lo menos un algoritmos o IDE diferente.

En el análisis de varianza de un factor, se empleó la probabilidad de noventa y cinco por ciento de confiabilidad y cinco por ciento de error. Para aceptar las hipótesis, la probabilidad debe ser $p > 0,05$ mientras que con un valor inferior se rechazaba y era necesario comparar las diferencias de medias con la prueba Tukey, para realizar dicha prueba se utilizó IBM SPSS que es un programa estadístico que permite realizar procesos analíticos, esta prueba sirvió para comprobar las diferencias entre las medias y obtener los grupos que se forman a partir de dichas diferencias, previo a realización de la prueba Tukey se calculó la media aritmética del grupo de datos de los IDE y algoritmos correspondientes.

TALENTO HUMANO

En el presente Trabajo de Titulación intervinieron los siguientes participantes:

- Postulantes :Carmen G. Cedeño Cool y Diana R. López Cedeño
- Tutor: Ing. Luis Cristóbal Cedeño Valarezo, MGTR.

Cabe recalcar que el presente Trabajo de Titulación no tuvo aportes económicos durante los procesos de intervención.

CAPÍTULO III. DESCRIPCIÓN DE LA EXPERIENCIA

El desarrollo del presente trabajo de titulación forma parte de un proyecto de investigación del Ing. Luis Cedeño Valarezo, titulado: Procesamiento de textos con gran cantidad de datos, donde las autoras aportan en: Determinar la eficiencia de los algoritmos de búsqueda de patrones en textos, para establecer cuál o cuáles presentan un mejor comportamiento. Con esta intervención se obtuvo el análisis de los algoritmos, así, como los resultados obtenidos del estudio de la eficiencia y eficacia de los mismos.

A continuación, se detallan los resultados obtenidos en cada objetivo:

3.1. DEFINIR LOS ALGORITMOS DE BÚSQUEDA DE PATRONES DE TEXTO

El primer objetivo de la investigación se centró en la búsqueda de los algoritmos de búsqueda de patrones de texto, como lo muestra el Cuadro 2.1 y posteriormente se definió mediante una lista los algoritmos a analizar, tal como consta en el (Anexo 1).

3.2. ANALIZAR LOS ALGORITMOS

En el estudio del funcionamiento de los algoritmos de búsqueda de patrones de texto se empleó una ficha de observación (Anexo 2).

Para determinar la eficacia de los algoritmos se empleó el indicador: cantidad de aciertos, así mismo, para determinar la eficiencia se utilizaron los indicadores: recursos para almacenar (RAM), recursos para procesar (RAM) y tiempo de respuesta la cual permitió registrar los recursos empleados por cada algoritmo (Anexo 3).

Como se mencionó en el capítulo anterior los algoritmos: Fuerza bruta, Boyer-Moore Horspool y Knuth-Morris-Pratt, fueron ejecutados en computadoras con diferentes características.

El primer equipo de cómputo en el que fueron analizados los algoritmos tuvo las siguientes características:

Cuadro 3. 1. Resultados de los algoritmos compilados en C-Free - computadora 1

Computadora: 1	-	Sistema Operativo: Windows 8.1 Pro		
	-	Procesador: Pentium® Dual-Core CPU E5200 @ 2.50 GHz 2.50 GHz		
	-	Memoria instalada (RAM): 2,00 GB		
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64		
C-Free				
	Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73	23,0	1,1	0,073
Boyer-Moore- Horspool	1 de 73	22,7	1,1	0,014
Knuth-Morris-Pratt	14 de 73	23,3	1,1	0,032

A continuación se presentan los resultados obtenido en C-Free:

En los resultados generados por el algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto fue: 73 de 73, mientras que los indicadores de eficiencia: recursos para almacenar (RAM) señaló un consumo de 23,0 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,073 ms.

Por otra parte, el algoritmo Boyer-Moore-Horspool emitió los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 22,7 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta fue de: 0,014 ms.

En el algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 23,3 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,032 ms.

Cuadro 3. 2. Resultados de los algoritmos compilados en Dev-C++ - computadora 1

Computadora: 1	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Pentium® Dual-Core CPU E5200 @ 2.50 GHz 2.50 GHz			
	-	Memoria instalada (RAM): 2,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	8,6	1,1	0,093	
Boyer-Moore- Horspool	1 de 73	8,8	1,1	0,018	
Knuth-Morris-Pratt	73 de 73	9,1	1,1	0,141	

Según los resultados obtenidos al compilar los algoritmos en Dev-C++ se obtuvo lo siguiente:

Con respecto al algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto fue: 73 de 73, mientras que en recursos para almacenar (RAM) señaló un consumo de 8,6 MB, en recursos para procesar (RAM) 1,1 MB y el tiempo de fue: 0,093 ms.

El algoritmo Boyer-Moore-Horspool obtuvo los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 8,8 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,018 ms.

Los resultados del algoritmo Knuth-Morris-Pratt: cantidad de aciertos encontrados: 73 de 73, recursos para almacenar (RAM): 9,1 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta fue de: 0,141 ms.

Cuadro 3. 3. Resultados de los algoritmos compilados en CodeBlocks - computadora 1

Computadora: 1	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Pentium® Dual-Core CPU E5200 @ 2.50 GHz 2.50 GHz			
	-	Memoria instalada (RAM): 2,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	77,3	1,1	0,094	
Boyer-Moore- Horspool	1 de 73	77,8	1,1	0,000	
Knuth-Morris-Pratt	14 de 73	76,9	1,1	0,047	

Y de acuerdo a los resultados al compilar los algoritmos en CodeBlocks se obtuvo lo siguiente:

Los resultados del algoritmo Fuerza Bruta fueron los siguientes: cantidad de aciertos encontrados en el texto: 73 de 73, recursos para almacenar (RAM): 77,3 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,094 ms.

Así mismo el algoritmo Boyer-Moore-Horspool obtuvo los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 77,8 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta: 0,00 ms.

El algoritmo Knuth-Morris-Pratt en la cantidad de aciertos encontrados obtuvo los siguientes resultados: 14 de 73 aciertos, recursos para almacenar (RAM): 76,9 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta fue de: 0,047 ms.

Las características del segundo equipo de cómputo fueron las siguientes:

Cuadro 3. 4. Resultados de los algoritmos compilados en C-Free - computadora 2

Computadora: 2	-	Sistema Operativo: Windows 8.1 Pro		
	-	Procesador: Intel® Core™ i3-3240 CPU @ 3.40GHz 3.40 GHz		
	-	Memoria instalada (RAM): 8,00 GB (7,89 GB utilizable)		
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64		
C-Free				
Algoritmos	Indicador de eficacia	Indicadores de eficiencia		
	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73	23,1	1,0	0,031
Boyer-Moore-Horspool	1 de 73	22,9	1,1	0,023
Knuth-Morris-Pratt	14 de 73	23,2	1,0	0,016

A continuación se presentan los resultados obtenidos en C-Free:

En los resultados generados por el algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto, fue de 73 de 73 aciertos, recursos para almacenar (RAM): 23,1 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,031 ms.

El algoritmo Boyer-Moore-Horspool presentó los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 22,9 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,023 ms.

Y el algoritmo Knuth-Morris-Pratt obtuvo los siguientes resultados: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 23,2 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,016 ms.

Cuadro 3. 5. Resultados de los algoritmos compilados en Dev-C++ - computadora 2

Computadora: 2	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core™ i3-3240 CPU @ 3.40GHz 3.40 GHz			
	-	Memoria instalada (RAM): 8,00 GB (7,89 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	28,3	0,8	0,031	
Boyer-Moore-Horspool	1 de 73	27,7	1,0	0,030	
Knuth-Morris-Pratt	73 de 73	28,5	0,8	0,047	

Al compilar los algoritmos en Dev-C++ se obtuvieron los siguientes resultados:

La cantidad de aciertos encontrados en el texto en el algoritmo Fuerza Bruta, fue: 73 de 73 aciertos, recursos para almacenar (RAM): 28,3 MB, en recursos para procesar (RAM): 0,8 MB y el tiempo de respuesta: 0,031 ms.

En el caso del algoritmo Boyer-Moore-Horspool, la cantidad de aciertos encontrados fue: 1 de 73, recursos para almacenar (RAM): 27,7 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,030 ms.

Por otra parte en el algoritmo Knuth-Morris-Pratt, la cantidad de aciertos encontrados fue: 73 de 73, recursos para almacenar (RAM): 28,5 MB, en recursos para procesar (RAM): 0,8 MB y el tiempo de respuesta fue de: 0,047 ms.

Cuadro 3. 6. Resultados de los algoritmos compilados en CodeBlocks - computadora 2

Computadora: 2	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core™ i3-3240 CPU @ 3.40GHz 3.40 GHz			
	-	Memoria instalada (RAM): 8,00 GB (7,89 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	75,9	1,1	0,031	
Boyer-Moore-Horspool	1 de 73	77,4	1,0	0,000	
Knuth-Morris-Pratt	14 de 73	76,2	1,1	0,015	

En CodeBlocks se obtuvieron los siguientes resultados:

En el algoritmo Fuerza Bruta la cantidad de aciertos encontrados fue: 73 de 73 aciertos, recursos para almacenar (RAM): 75,9 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta fue de: 0,031 ms.

Mientras que el algoritmo Boyer-Moore-Horspool en la cantidad de aciertos encontró: 1 de 73, recursos para almacenar (RAM): 77,4 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que empleó fue: 0,00 ms.

En el caso del algoritmo Knuth-Morris-Pratt, la cantidad de aciertos encontrados fue: 14 de 73, recursos para almacenar (RAM): 76,2 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,015 ms.

El tercer equipo de cómputo en el que fueron analizados los algoritmos tenía las siguientes características:

Cuadro 3. 7. Resultados de los algoritmos compilados en C-Free - computadora 3

Computadora: 3	-	Sistema Operativo: Windows 7 Professional			
	-	Procesador: Intel® Core™ i3 CPU 530 @ 2.93GHz 2.93			
	-	Memoria instalada (RAM): 4,00 GB (3,05 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits			
C-Free					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	0,03	0,002	0,076	
Boyer-Moore-Horspool	1 de 73	0,03	0,002	0,053	
Knuth-Morris-Pratt	12 de 73	0,02	0,002	0,018	

Con referencia a lo anterior se presentan los resultados obtenido en C-Free:

En cuanto a los resultados generados por el algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto, fue de 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM): 0,03 MB, en recursos para procesar (RAM): 0,002 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,076 ms.

Por otro lado, el algoritmo Boyer-Moore-Horspool emitió los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 0,03 MB, en recursos para procesar (RAM): 0,002 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,053 ms.

De igual manera, en el algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 12 de 73, recursos para almacenar (RAM): 0,02 MB, en recursos para procesar (RAM): 0,002 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,018 ms.

Cuadro 3. 8. Resultados de los algoritmos compilados en Dev-C++ - computadora 2

Computadora: 3	-	Sistema Operativo: Windows 7 Professional			
	-	Procesador: Intel® Core™ i3 CPU 530 @ 2.93GHz 2.93			
	-	Memoria instalada (RAM): 4,00 GB (3,05 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	0,03	0,001	0,055	
Boyer-Moore-Horspool	1 de 73	0,03	0,001	0,030	
Knuth-Morris-Pratt	73 de 73	0,03	0,001	0,123	

Los resultados al compilar los algoritmos en Dev-C++ fueron:

En cuanto al algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto, fue: 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM) señaló un consumo de 0,03 MB, en recursos para procesar (RAM): 0,001 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,055 ms.

Por lo contrario, en el algoritmo Boyer-Moore-Horspool se consiguieron los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 0,03 MB, en recursos para procesar (RAM): 0,001 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,030 ms.

Sin embargo los resultados del algoritmo Knuth-Morris-Pratt fueron: cantidad de aciertos encontrados: 73 de 73, recursos para almacenar (RAM): 0,03 MB, en recursos para procesar (RAM): 0,001 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,123 ms.

Cuadro 3. 9. Resultados de los algoritmos compilados en CodeBlocks - computadora 3

Computadora: 3	-	Sistema Operativo: Windows 7 Professional			
	-	Procesador: Intel® Core™ i3 CPU 530 @ 2.93GHz 2.93			
	-	Memoria instalada (RAM): 4,00 GB (3,05 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	0,08	0,0009	0,063	
Boyer-Moore-Horspool	1 de 73	0,08	0,92	0,000	
Knuth-Morris-Pratt	14 de 73	0,08	0,0009	0,02	

En CodeBlocks se obtuvieron los siguientes resultados:

El algoritmo Fuerza Bruta en la cantidad de aciertos encontrados en el texto, mostró: 73 de 73 aciertos, recursos para almacenar (RAM): 0,08 MB, en recursos para procesar (RAM): 0,0009 MB y el tiempo de respuesta: 0,063 ms.

Sin embargo en el algoritmo Boyer-Moore-Horspool se lograron los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 0,08 MB, en recursos para procesar (RAM): 0,92 MB y el tiempo de respuesta: 0,00 ms.

Los resultados del algoritmo Knuth-Morris-Pratt en la cantidad de aciertos encontrados fue: 14 de 73, recursos para almacenar (RAM): 0,08 MB, en recursos para procesar (RAM): 0,0009 MB y el tiempo de respuesta fue: de 0,02 ms.

Con respecto a las características del cuarto equipo de cómputo en el que fueron analizados los algoritmos, se presenta la siguiente información:

Cuadro 3. 10. Resultados de los algoritmos compilados en C-Free - computadora 4

Computadora: 4	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core (TM) i5-3470 CPU @ 3.20 GHz 3.20 GHz			
	-	Memoria instalada (RAM): 6,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
C-Free					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos		Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73		23,2	1,1	0,032
Boyer-Moore-Horspool	1 de 73		22,8	1,1	0,024
Knuth-Morris-Pratt	14 de 73		23,2	1,1	0,015

Así mismo se presentan los resultados obtenido en C-Free:

En cuanto a los resultados generados por el algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto, fue de 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM): 23,2 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,032 ms.

En cambio el algoritmo Boyer-Moore-Horspool presentó los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 22,8 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,024 ms.

De igual forma en el algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 23,2 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,015 ms.

Cuadro 3. 11. Resultados de los algoritmos compilados en Dev-C++ - computadora 4

Computadora: 4	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core (TM) i5-3470 CPU @ 3.20 GHz 3.20 GHz			
	-	Memoria instalada (RAM): 6,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	9,3	1,1	0,031	
Boyer-Moore-Horspool	1 de 73	9,5	1,1	0,028	
Knuth-Morris-Pratt	73 de 73	9,4	1,1	0,046	

Los siguientes resultados corresponden a la compilación de los algoritmos en Dev-C++:

El algoritmo Fuerza Bruta presentó los siguientes resultados: la cantidad de aciertos encontrados en el texto fue: 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM) señaló un consumo de 9,3 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,031 ms.

En el algoritmo Boyer-Moore-Horspool se consiguieron los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 9,5 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,028 ms.

De modo similar, los resultados del algoritmo Knuth-Morris-Pratt fueron: cantidad de aciertos encontrados: 73 de 73, recursos para almacenar (RAM): 9,4 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,046 ms.

Cuadro 3. 12. Resultados de los algoritmos compilados en CodeBlocks - computadora 4

Computadora: 4	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core (TM) i5-3470 CPU @ 3.20 GHz 3.20 GHz			
	-	Memoria instalada (RAM): 6,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos		Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73		76,5	1,1	0,031
Boyer-Moore-Horspool	1 de 73		77,4	1,1,	0,000
Knuth-Morris-Pratt	14 de 73		76,1	1,1,	0,016

Tras la compilación de los algoritmos en CodeBlocks se lograron los siguientes resultados:

A través del algoritmo Fuerza Bruta se obtuvo lo siguiente: cantidad de aciertos encontrados: 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM): 76,5 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta: 0,031 ms.

Con el algoritmo Boyer-Moore-Horspool se consiguieron los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 77,4 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta fue de: 0,00 ms.

Y mediante el algoritmo Knuth-Morris-Pratt la cantidad de aciertos encontrados fueron: 14 de 73, recursos para almacenar (RAM): 76,1 MB, en recursos para procesar (RAM): 1,1 MB y el tiempo de respuesta: 0,016 ms.

El quinto equipo de cómputo cuenta con las siguientes características:

Cuadro 3. 13. Resultados de los algoritmos compilados en C-Free - computadora 5

Computadora: 5	-	Sistema Operativo: Windows 10		
	-	Procesador: Intel® Core™ i5-4210U CPU @1.70GHz 2.40 GHz		
	-	Memoria instalada (RAM): 8,00 GB		
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64		
C-Free				
Algoritmos	Indicador de eficacia	Indicadores de eficiencia		
	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73	23,9	0,5	0,124
Boyer-Moore-Horspool	7 de 73	23,7	0,4	0,030
Knuth-Morris-Pratt	62 de 73	25,4	0,5	0,203

Los resultados que se muestran a continuación fueron obtenidos en C-Free:

El algoritmo Fuerza Bruta en su indicador cantidad de aciertos encontrados en el texto (eficacia) fue de 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM): 23,9 MB, en recursos para procesar (RAM): 0,5 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,124 ms.

Los resultados del algoritmo Boyer-Moore-Horspool son: cantidad de aciertos encontrados: 7 de 73, recursos para almacenar (RAM): 23,7 MB, en recursos para procesar (RAM): 0,4 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,030 ms.

Según el análisis del algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 62 de 73, recursos para almacenar (RAM): 25,4 MB, en recursos para procesar (RAM): 0,5 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,203 ms.

Cuadro 3. 14. Resultados de los algoritmos compilados en Dev-C++ - computadora 5

Computadora: 5	-	Sistema Operativo: Windows 10			
	-	Procesador: Intel® Core™ i5-4210U CPU @1.70GHz 2.40 GHz			
	-	Memoria instalada (RAM): 8,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos		Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms
Fuerza Bruta	73 de 73		25,3	4,9	0,125
Boyer-Moore-Horspool	1 de 73		28,1	4,9	0,050
Knuth-Morris-Pratt	23 de 73		28,5	4,9	0,062

Sin embargo, en Dev-C++ se obtuvieron los siguientes resultados:

El algoritmo Fuerza Bruta en su indicador cantidad de aciertos encontrados en el texto fue: 73 de 73 aciertos, mientras que los indicadores de eficiencia: recursos para almacenar (RAM) señaló un consumo de 25,3 MB, en recursos para procesar (RAM): 4,9 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,125 ms.

Los resultados del algoritmo Boyer-Moore-Horspool son: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 28,1 MB, en recursos para procesar (RAM): 4,9 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,050 ms.

En el algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 23 de 73, recursos para almacenar (RAM): 28,5 MB, en recursos para procesar (RAM): 4,9 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,062 ms.

Cuadro 3. 15. Resultados de los algoritmos compilados en CodeBlocks - computadora 5

Computadora: 5	-	Sistema Operativo: Windows 10			
	-	Procesador: Intel® Core™ i5-4210U CPU @1.70GHz 2.40 GHz			
	-	Memoria instalada (RAM): 8,00 GB			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	82,8	4,9	0,156	
Boyer-Moore-Horspool	1 de 73	82,7	4,9	0,000	
Knuth-Morris-Pratt	12 de 73	82,4	4,9	0,032	

Con respecto a la compilación de los algoritmos en CodeBlocks se obtuvo lo siguiente:

En cuanto al algoritmo Fuerza Bruta, la cantidad de aciertos encontrados fue: 73 de 73 aciertos, recursos para almacenar (RAM): 82,8 MB, en recursos para procesar (RAM): 4,9 MB y el tiempo de respuesta fue: 0,156 ms.

Los resultados del algoritmo Boyer-Moore-Horspool fueron los siguientes: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 82,7 MB, en recursos para procesar (RAM): 4,9 MB y el tiempo de respuesta: 0,00 ms.

Y en el algoritmo Knuth-Morris-Pratt se obtuvieron los siguientes resultados: cantidad de aciertos encontrados: 12 de 73, recursos para almacenar (RAM): 82,4 MB, en recursos para procesar (RAM): 4,9 MB y tiempo de respuesta: 0,032 ms.

A continuación se detallan las características del sexto equipo de cómputo donde fueron analizados los algoritmos:

Cuadro 3. 16. Resultados de los algoritmos compilados en C-Free - computadora 6

Computadora: 6	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core™ i5-3330 CPU @ 3.00GHz 3.00 GHz			
	-	Memoria instalada (RAM): 4,00 GB (3,88 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
C-Free					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta MB	
Fuerza Bruta	73 de 73	24,6	1,0	0,032	
Boyer-Moore-Horspool	1 de 73	24,0	1,0	0,000	
Knuth-Morris-Pratt	14 de 73	24,7	1,0	0,015	

Según lo antes mencionados, se presentan los resultados obtenidos en C-Free:

Los resultados generados por el algoritmo Fuerza Bruta fueron: la cantidad de aciertos encontrados en el texto, fue de 73 de 73 aciertos, en recursos para almacenar (RAM): 24,6 MB, recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta: 0,032 ms.

En lo que respecta al algoritmo Boyer-Moore-Horspool se presentan los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 24,0 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,0 ms.

En el algoritmo Knuth-Morris-Pratt los resultados fueron: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 24,7 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,015 ms.

Cuadro 3. 17. Resultados de los algoritmos compilados en Dev-C++ - computadora 6

Computadora: 6	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core™ i5-3330 CPU @ 3.00GHz 3.00 GHz			
	-	Memoria instalada (RAM): 4,00 GB (3,88 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
Dev-C++					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	8,9	1,0	0,032	
Boyer-Moore-Horspool	1 de 73	28,1	1,0	0,029	
Knuth-Morris-Pratt	14 de 73	9,0	1,0	0,015	

Por otra parte los resultados de la compilación de los algoritmos en Dev-C++ son los siguientes:

Con respecto al algoritmo Fuerza Bruta, la cantidad de aciertos encontrados en el texto, fue de 73 de 73 aciertos, recursos para almacenar (RAM): de 8,9 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta fue de: 0,032 ms.

Así mismo el algoritmo Boyer-Moore-Horspool mostró los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 28,1 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que empleó el algoritmo en mostrar el resultado fue de 0,029 ms.

Finalmente el algoritmo Knuth-Morris-Pratt emitió lo siguientes resultados: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 9,0 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta que utilizó el algoritmo en mostrar el resultado fue de 0,015 ms.

Cuadro 3. 18. Resultados de los algoritmos compilados en CodeBlocks - computadora 6

Computadora: 6	-	Sistema Operativo: Windows 8.1 Pro			
	-	Procesador: Intel® Core™ i5-3330 CPU @ 3.00GHz 3.00 GHz			
	-	Memoria instalada (RAM): 4,00 GB (3,88 GB utilizable)			
	-	Tipo de Sistema: Sistema operativo de 64 bits, procesador x64			
CodeBlocks					
		Indicador de eficacia	Indicadores de eficiencia		
Algoritmos	Cantidad de aciertos	Para almacenar (RAM) MB	Para procesar (RAM) MB	Tiempo de respuesta ms	
Fuerza Bruta	73 de 73	77,1	1,0	0,042	
Boyer-Moore-Horspool	1 de 73	75,9	1,0	0,005	
Knuth-Morris-Pratt	14 de 73	75,8	1,0	0,016	

Por otro lado los resultados de la compilación de los algoritmos en CodeBlocks fueron:

El algoritmo Fuerza Bruta mostró los siguientes resultados: cantidad de aciertos encontrados: 73 de 73 aciertos, recursos para almacenar (RAM): 77,1 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta: 0,042 ms.

Mientras que el algoritmo Boyer-Moore-Horspool mostró los siguientes resultados: cantidad de aciertos encontrados: 1 de 73, recursos para almacenar (RAM): 75,9 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta: 0,005 ms.

Y el algoritmo Knuth-Morris-Pratt emitió los siguientes resultados: cantidad de aciertos encontrados: 14 de 73, recursos para almacenar (RAM): 75,8 MB, en recursos para procesar (RAM): 1,0 MB y el tiempo de respuesta: 0,016 ms.

Cuadro 3. 19. Resultado del análisis del algoritmo MapReduce en Google Cloud Platform.

Google Cloud Platform			
	Indicador de eficacia	Indicadores de eficiencia	
Algoritmo	Cantidad de aciertos filas	Para procesar (RAM) GB	Tiempo de respuesta s
MapReduce	6 de 1340811	6,79	19,3

De acuerdo a los resultados del algoritmo MapReduce la cantidad de filas analizadas fue de 1340811 donde se encontraron 6 coincidencias, empelando 6,79 GB en procesamiento y en un tiempo de respuesta: 19,3 s

3.3. ESTABLECER LA EFICIENCIA Y EFICACIA DE LOS MISMOS

El cumplimiento de la primera actividad permitió obtener el cuadro comparativo de los algoritmos de búsqueda de patrones de texto, con los IDE empleados y sus indicadores.

Cuadro 3. 20. Cuadro comparativo de los algoritmos e IDE.

Algoritmo	C-Free				Dev-C++				CodeBlocks			
	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos
Boyer-Moore Horspool	0,014	22,700	1,100	0,014	0,014	8,800	1,100	0,018	0,014	77,800	1,100	0,000
	0,014	22,900	1,100	0,023	0,014	27,700	1,000	0,030	0,014	77,400	1,000	0,000
	0,014	24,664	1,992	0,053	0,014	29,551	1,457	0,030	0,014	82,816	0,926	0,000
	0,014	22,800	1,100	0,024	0,014	9,500	1,100	0,028	0,014	77,400	1,100	0,000
	0,096	23,700	0,400	0,030	0,014	28,100	4,900	0,050	0,014	82,700	4,900	0,000
	0,014	24,000	1,000	0,000	0,014	28,100	1,000	0,029	0,014	75,900	1,000	0,005
Fuerza Bruta	1,000	23,000	1,100	0,073	1,000	8,600	1,100	0,093	1,000	77,300	1,100	0,094
	1,000	23,100	1,000	0,031	1,000	28,300	0,800	0,031	1,000	75,900	1,100	0,031
	1,000	24,566	1,988	0,076	1,000	29,934	1,461	0,055	1,000	77,199	0,992	0,063
	1,000	23,200	1,100	0,032	1,000	9,300	1,100	0,031	1,000	76,500	1,100	0,031
	1,000	23,900	0,500	0,124	1,000	25,300	4,900	0,125	1,000	82,800	4,900	0,156
	1,000	24,600	1,000	0,032	1,000	8,900	1,000	0,032	1,000	77,100	1,000	0,042
Knuth-Morris-Pratt	0,192	23,300	1,100	0,032	1,000	9,100	1,100	0,141	0,192	76,900	1,100	0,047
	0,192	23,200	1,000	0,016	1,000	28,500	0,800	0,047	0,192	76,200	1,100	0,015
	0,164	24,625	1,988	0,018	1,000	29,906	1,465	0,123	0,192	77,762	0,980	0,020
	0,192	23,200	1,100	0,015	1,000	9,400	1,100	0,046	0,192	76,100	1,100	0,016
	0,849	25,400	0,500	0,203	0,315	28,500	4,900	0,062	0,164	82,400	4,900	0,032
	0,192	24,700	1,000	0,015	0,192	9,000	1,000	0,015	0,192	75,800	1,000	0,016

Cuadro 3. 21. Cuadro comparativo de IDE y algoritmos.

IDE	Boyer-Moore Horspool				Fuerza Bruta				Knuth-Morris-Pratt			
	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos	Cantidad de aciertos	Almacenar (RAM) MB	Procesar (RAM) MB	Tiempo de respuesta en milisegundos
C-Free	0,014	22,700	1,100	0,014	1,000	23,000	1,100	0,073	0,192	23,300	1,100	0,032
	0,014	22,900	1,100	0,023	1,000	23,100	1,000	0,031	0,192	23,200	1,000	0,016
	0,014	24,664	1,992	0,053	1,000	24,566	1,988	0,076	0,164	24,625	1,988	0,018
	0,014	22,800	1,100	0,024	1,000	23,200	1,100	0,032	0,192	23,200	1,100	0,015
	0,096	23,700	0,400	0,030	1,000	23,900	0,500	0,124	0,849	25,400	0,500	0,203
	0,014	24,000	1,000	0,000	1,000	24,600	1,000	0,032	0,192	24,700	1,000	0,015
Dev C++	0,014	8,800	1,100	0,018	1,000	8,600	1,100	0,093	1,000	9,100	1,100	0,141
	0,014	27,700	1,000	0,030	1,000	28,300	0,800	0,031	1,000	28,500	0,800	0,047
	0,014	29,551	1,457	0,030	1,000	29,934	1,461	0,055	1,000	29,906	1,465	0,123
	0,014	9,500	1,100	0,028	1,000	9,300	1,100	0,031	1,000	9,400	1,100	0,046
	0,014	28,100	4,900	0,050	1,000	25,300	4,900	0,125	0,315	28,500	4,900	0,062
	0,014	28,100	1,000	0,029	1,000	8,900	1,000	0,032	0,192	9,000	1,000	0,015
CodeBlocks	0,014	77,800	1,100	0,000	1,000	77,300	1,100	0,094	0,192	76,900	1,100	0,047
	0,014	77,400	1,000	0,000	1,000	75,900	1,100	0,031	0,192	76,200	1,100	0,015
	0,014	82,816	0,926	0,000	1,000	77,199	0,992	0,063	0,192	77,762	0,980	0,020
	0,014	77,400	1,100	0,000	1,000	76,500	1,100	0,031	0,192	76,100	1,100	0,016
	0,014	82,700	4,900	0,000	1,000	82,800	4,900	0,156	0,164	82,400	4,900	0,032
	0,014	75,900	1,000	0,005	1,000	77,100	1,000	0,042	0,192	75,800	1,000	0,016

Cuadro 3. 22. Cuadro de resultados de MapReduce.

Google Cloud Platform			
	Indicador de eficacia	Indicadores de eficiencia	
Algoritmo	Cantidad de aciertos filas	Para procesar (RAM) GB	Cantidad de aciertos s
MapReduce	6 de 1340811	6,79	19,3

A continuación se presenta el diseño experimental con análisis de varianza de uno y dos factores:

3.3.1. CANTIDAD DE ACIERTOS

3.3.1.1. IDE

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 23. Análisis de varianza de dos factores de indicador cantidad de aciertos con la variable IDE.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Muestra	8,78810561	2	4,39405281	175,927178	5,3466E-22	3,20431729
Columnas	0,35158427	2	0,17579214	7,03828923	0,00218996	3,20431729
Interacción	0,7220589	4	0,18051472	7,22737012	0,00013893	2,57873918
Dentro del grupo	1,12394445	45	0,02497654			
Total	10,9856932	53				

Análisis de interacción: el nivel de interacción o relación que existe entre los algoritmos y los IDE es de 0,00013893.

Análisis de varianza de un factor

H0: Los IDE producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 24. Análisis de varianza de un factor de indicador cantidad de aciertos con la variable IDE.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Entre grupos	0,35158427	2	0,17579214	0,84307947	0,436292531	3,17879929
Dentro de los grupos	10,634109	51	0,20851194			
Total	10,9856932	53				

P= 0,4

Análisis de probabilidad: Se acepta H0. Es decir, los IDE producen resultados iguales.

3.3.1.2. ALGORITMOS

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 25. Análisis de varianza de dos factores de indicador cantidad de aciertos con la variable algoritmos.

ANÁLISIS DE VARIANZA							
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F	
Muestra	0,35158427	2	0,17579214	7,03828923	0,00218996	3,20431729	
Columnas	8,78810561	2	4,39405281	175,927178	5,3466E-22	3,20431729	
Interacción	0,7220589	4	0,18051472	7,22737012	0,00013893	2,57873918	
Dentro del grupo	1,12394445	45	0,02497654				
Total	10,9856932	53					

Análisis de interacción: el nivel de interacción o relación que existe entre los IDE y los algoritmos es de 0,00013893.

Análisis de varianza de un factor

H0: Los algoritmos producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 26. Análisis de varianza de un factor de indicador cantidad de aciertos con la variable algoritmos.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Entre grupos	8,78810561	2	4,39405281	101,973951	1,5084E-18	3,17879929
Dentro de los grupos	2,19758762	51	0,04308995			
Total	10,9856932	53				

P= 1,5084E-18

Análisis de probabilidad: Se rechaza H0, es decir, existe por lo menos uno que es diferente.

Prueba Tukey

Cuadro 3. 27. Media Aritmética de indicador cantidad de aciertos con la variable algoritmos.

Media Aritmética	
Algoritmos	Aciertos
Boyer-MooreHorspool	0,014
Boyer-MooreHorspool	0,014
Boyer-MooreHorspool	0,027
Knuth-Morris-Pratt	0,187
Knuth-Morris-Pratt	0,297
Knuth-Morris-Pratt	0,751
Fuerza Bruta	1,000
Fuerza Bruta	1,000
Fuerza Bruta	1,000

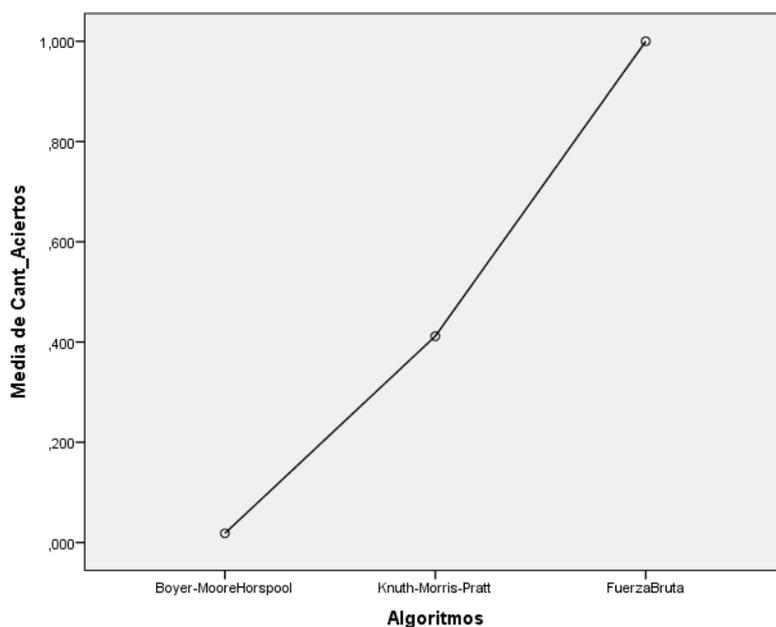
Cuadro 3. 28. Prueba Tukey indicador cantidad de aciertos con la variable algoritmos.

Cant_Aciertos		
HSD Tukey ^a		
Algoritmos	N	Subconjunto para alfa = 0.05
		1 2
Boyer-MooreHorspool	3	,01833
Knuth-Morris-Pratt	3	,41167
FuerzaBruta	3	1,00000
Sig.		,071 1,000

Se visualizan las medias para los grupos en los subconjuntos homogéneos.

a. Utiliza el tamaño de la muestra de la media armónica = 3,000.

Análisis Tukey: Según la Prueba de Tukey existen dos grupos, el primero está conformado por los algoritmos Boyer-MooreHorspool y Knuth-Morris-Pratt con un nivel de significancia de 0,71, mientras que en el segundo grupo se encuentra Fuerza Bruta con un nivel de significancia de 1,000. Para una mejor comprensión, se presenta el siguiente gráfico:

Gráfico 3. 1. Representación de medias de prueba Tukey de indicador cantidad de aciertos con la variable algoritmos.

3.3.2. ALMACENAMIENTO (RAM)

3.3.2.1. IDE

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 29. Análisis de varianza de dos factores de indicador almacenamiento con la variable IDE.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Muestra	23,3129375	2	11,6564687	0,29991976	0,74234705	3,20431729
Columnas	38218,1843	2	19109,0921	491,675005	2,6553E-31	3,20431729
Interacción	28,3258227	4	7,08145567	0,18220514	0,94645271	2,57873918
Dentro del grupo	1748,9381	45	38,865291			
Total	40018,7611	53				

Análisis de interacción: el nivel de interacción o relación que existe entre los algoritmos y los IDE es de 0,94645271.

Análisis de varianza de un factor

H0: Los IDE producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 30. Análisis de varianza de un factor de indicador almacenamiento con la variable IDE.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Entre grupos	38218,1843	2	19109,0921	541,25082	4,5214E-35	3,17879929
Dentro de los grupos	1800,57686	51	35,3054285			
Total	40018,7611	53				

P= 4,5214E-35

Análisis de probabilidad: Se rechaza H0, es decir existe por lo menos uno que es diferente.

Prueba Tukey

Cuadro 3. 31. Media Aritmética de indicador almacenamiento con la variable IDE.

Media Aritmetica	
IDE	Almacenamiento (RAM)
Dev-C++	18,39
Dev-C++	19,07
Dev-C++	21,96
C-Free	23,46
C-Free	23,73
C-Free	24,07
CodeBlocks	77,53
CodeBlocks	77,8
CodeBlocks	79,00

Cuadro 3. 32. Prueba Tukey indicador almacenamiento con la variable IDE.

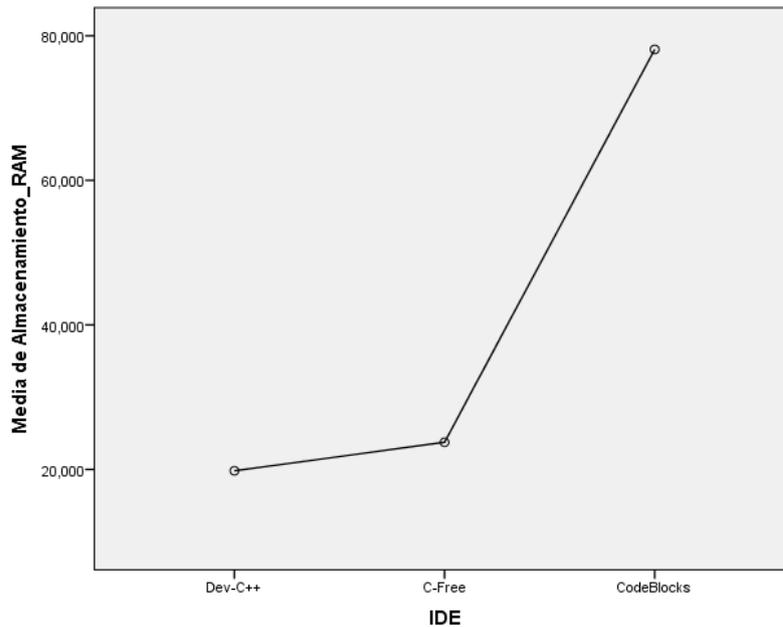
Almacenamiento_RAM				
HSD Tukey ^a				
IDE	N	Subconjunto para alfa = 0.05		
		1	2	3
Dev-C++	3	19,80667		
C-Free	3		23,75333	
CodeBlocks	3			78,11000
Sig.		1,000	1,000	1,000

Se visualizan las medias para los grupos en los subconjuntos homogéneos.

a. Utiliza el tamaño de la muestra de la media armónica = 3,000.

Análisis Tukey: Según la Prueba de Tukey existen tres grupos, el primero está conformado por el IDE Dev-C++ con un nivel de significancia de 1,000, mientras que el segundo grupo corresponde al IDE C-Free con un nivel de significancia de 1,000 y en el tercer grupo se encuentra CodeBlocks con un nivel de significancia de 1,000. Los datos antes descritos se pueden apreciar en siguiente gráfico:

Gráfico 3. 2. Representación de medias de prueba Tukey de indicador almacenamiento con la variable IDE.



3.3.2.2. ALGORITMOS

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 33. Análisis de varianza de dos factores de indicador almacenamiento con la variable algoritmos.

ANÁLISIS DE VARIANZA							
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F	
Muestra	38218,1843	2	19109,0921	491,675005	2,6553E-31	3,20431729	
Columnas	23,3129375	2	11,6564687	0,29991976	0,74234705	3,20431729	
Interacción	28,3258227	4	7,08145567	0,18220514	0,94645271	2,57873918	
Dentro del grupo	1748,9381	45	38,865291				
Total	40018,7611	53					

Análisis de interacción: el nivel de interacción o relación que existe entre las IDE y los algoritmos es de 0,94645271.

Análisis de varianza de un factor

H0: Los algoritmos producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 34. Análisis de varianza de un factor de indicador almacenamiento con la variable algoritmos.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Entre grupos	23,3129375	2	11,6564687	0,01486369	0,9852505	3,17879929
Dentro de los grupos	39995,4482	51	784,224474			
Total	40018,7611	53				

P= 0,9

Análisis de probabilidad: Se acepta H0. Es decir, los algoritmos producen resultados iguales.

3.3.3. PROCESAMIENTO

3.3.3.1. IDE

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 35. Análisis de varianza de dos factores de indicador procesamiento con la variable IDE.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Muestra	5,3779E-05	2	2,6889E-05	1,5632E-05	0,99998437	3,204317292
Columnas	4,31925483	2	2,15962741	1,25552483	0,29471517	3,204317292
Interacción	0,0070124	4	0,0017531	0,00101918	0,99999783	2,578739184
Dentro del grupo	77,4044696	45	1,72009932			
Total	81,7307906	53				

Análisis de interacción: el nivel de interacción o relación que existe entre los algoritmos y las IDE es de 0,99999783.

Análisis de varianza de un factor

H0: Las IDE producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 36. Análisis de varianza de un factor de indicador procesamiento con la variable IDE.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Entre grupos	4,31925483	2	2,15962741	1,42279826	0,25044339	3,17879929
Dentro de los grupos	77,4115358	51	1,51787325			
Total	81,7307906	53				

P= 0,2

Análisis de probabilidad: Se acepta H0. Es decir, las IDE producen resultados iguales.

3.3.3.2. ALGORITMOS

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 37. Análisis de varianza de dos factores de indicador procesamiento con la variable algoritmos.

ANÁLISIS DE VARIANZA						
Origen de las variaciones	Suma de cuadrados	Grados de libertad	Promedio de los cuadrados	F	Probabilidad	Valor crítico para F
Muestra	5,3779E-05	2	2,6889E-05	1,5632E-05	0,99998437	3,20431729
Columnas	4,31925483	2	2,15962741	1,25552483	0,29471517	3,20431729
Interacción	0,0070124	4	0,0017531	0,00101918	0,99999783	2,57873918
Dentro del grupo	77,4044696	45	1,72009932			
Total	81,7307906	53				

Análisis de interacción: el nivel de interacción o relación que existe entre las IDE y los algoritmos es de 0,99999783.

Análisis de varianza de un factor

H0: Los algoritmos producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 38. Análisis de varianza de un factor de indicador procesamiento con la variable algoritmos.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Entre grupos	4,31925483	2	2,15962741	1,42279826	0,25044339	3,17879929
Dentro de los grupos	77,4115358	51	1,51787325			
Total	81,7307906	53				

P= 0,2

Análisis de probabilidad: Se acepta H0. Es decir, los algoritmos producen resultados iguales.

3.3.4. TIEMPO DE RESPUESTA

3.3.4.1. IDE

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 39. Análisis de varianza de dos factores de indicador tiempo de respuesta con la variable IDE.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Muestra	0,01927181	2	0,00963591	6,20578575	0,00416672	3,20431729
Columnas	0,00489626	2	0,00244813	1,57666189	0,21786652	3,20431729
Interacción	0,00526385	4	0,00131596	0,84751584	0,50264435	2,57873918
Dentro del grupo	0,06987283	45	0,00155273			
Total	0,09930476	53				

Análisis de interacción: el nivel de interacción o relación que existe entre los algoritmos y las IDE es de 0,50264435.

Análisis de varianza de un factor

H0: Las IDE producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 40. Análisis de varianza de un factor de indicador tiempo de respuesta con la variable IDE.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Entre grupos	0,00489626	2	0,00244813	1,32249333	0,27545229	3,17879929
Dentro de los grupos	0,0944085	51	0,00185115			
Total	0,09930476	53				

P= 0,2

Análisis de probabilidad: Se acepta H0. Es decir, las IDE producen resultados iguales.

3.3.4.2. ALGORITMOS

Análisis de varianza de dos factores con varias muestras por grupo

Cuadro 3. 41. Análisis de varianza de dos factores de indicador tiempo de respuesta con la variable algoritmos.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Muestra	0,00489626	2	0,00244813	1,57666189	0,21786652	3,20431729
Columnas	0,01927181	2	0,00963591	6,20578575	0,00416672	3,20431729
Interacción	0,00526385	4	0,00131596	0,84751584	0,50264435	2,57873918
Dentro del grupo	0,06987283	45	0,00155273			
Total	0,09930476	53				

Análisis de interacción: el nivel de interacción o relación que existe entre las IDE y los algoritmos es de 0,50264435.

Análisis de varianza de un factor

H0: Los algoritmos producen resultados iguales

Vs

H1: Existe por lo menos uno que es diferente

Cuadro 3. 42. Análisis de varianza de un factor de indicador tiempo de respuesta con la variable algoritmos.

ANÁLISIS DE VARIANZA						
<i>Origen de las variaciones</i>	<i>Suma de cuadrados</i>	<i>Grados de libertad</i>	<i>Promedio de los cuadrados</i>	<i>F</i>	<i>Probabilidad</i>	<i>Valor crítico para F</i>
Entre grupos	0,01927181	2	0,00963591	6,14036234	0,0040796	3,17879929
Dentro de los grupos	0,08003294	51	0,00156927			
Total	0,09930476	53				

P= 0,004

Análisis de probabilidad: Se rechaza H0, es decir, existe por lo menos uno que es diferente.

Prueba Tukey

Cuadro 3. 43. Media Aritmética de indicador tiempo de respuesta con la variable algoritmos.

Media Aritmetica	
Algoritmos	Tiempo de Respuesta
Boyer-MooreHorspool	0,00
Boyer-MooreHorspool	0,02
Knuth-Morris-Pratt	0,02
Boyer-MooreHorspool	0,03
Knuth-Morris-Pratt	0,05
Fuerza Bruta	0,06
Fuerza Bruta	0,06
Knuth-Morris-Pratt	0,07
Fuerza Bruta	0,07

Cuadro 3. 44. Prueba Tukey indicador tiempo de respuesta con la variable algoritmos.

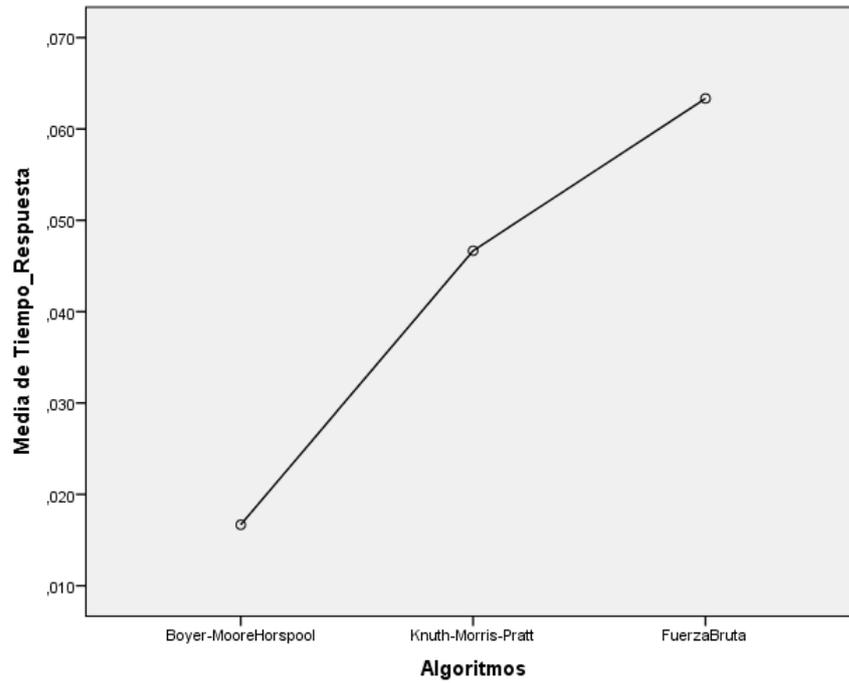
Tiempo Respuesta			
HSD Tukey ^a			
Algoritmos	N	Subconjunto para alfa = 0.05	
		1	2
Boyer-MooreHorspool	3	,01667	
Knuth-Morris-Pratt	3	,04667	,04667
FuerzaBruta	3		,06333
Sig.		,165	,507

Se visualizan las medias para los grupos en los subconjuntos homogéneos.

a. Utiliza el tamaño de la muestra de la media armónica = 3,000.

Análisis Tukey: Según la Prueba de Tukey existen dos grupos, el primero está conformado por los algoritmos Boyer-MooreHorspool y Knuth-Morris-Pratt con un nivel de significancia de 0,165, mientras que en el segundo grupo se encuentra Knuth-Morris-Pratt y Fuerza Bruta con un nivel de significancia de 0,507, dicha información se puede apreciar en el siguiente gráfico:

Gráfico 3. 3. Representación de medias de prueba Tukey de indicador tiempo de respuesta con la variable algoritmos.



CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- De acuerdo al método bibliográfico se pudo llegar a conocer cuáles son los algoritmos más utilizados en la búsqueda de patrones de texto.
- Por medio del método analítico se realizó el estudio de los algoritmos con las IDE C-Free, Dev-C++ y CodeBlocks, para ello se hizo uso de 6 computadores con diferentes características en la que se emplearon indicadores de eficiencia, y eficacia, lo cual permitió conocer que el algoritmo que más aciertos encuentra al ser analizado en diferentes computadoras es fuerza bruta, mientras que el algoritmo MapReduce permitió ser analizado una vez debido a que solo se usa en la nube.
- Finalmente, a través del diseño experimental se estableció que en los indicadores: cantidad de aciertos y tiempo de respuesta, Fuerza Bruta fue el algoritmo más eficiente y eficaz, mientras que en el indicador almacenamiento, el compilador que mejor se comportó con los algoritmos fue CodeBlocks.

4.2. RECOMENDACIONES

- En la búsqueda de patrones de texto se recomienda utilizar los algoritmos Fuerza Bruta, Boyer-Moore-Horspool, Knuth-Morris-Pratt y MapReduce ya que son los más usados.
- Se recomienda realizar el análisis de los algoritmos en diferentes computadoras para observar el comportamiento que presenta cada uno de ellos, sin embargo, el algoritmo MapReduce sería una excepción ya que éste fue implementado para ser usado exclusivamente en la nube.
- Es recomendable aumentar el tamaño de muestra en la cantidad de aciertos y tiempo de respuesta de los algoritmos y en el recurso almacenamiento (RAM) de las IDE para reducir el margen de error.

BIBLIOGRAFÍA

- Abreu, J. 2014. El método de la investigación. Monterrey, MX. Revista Daena: International Journal of Good Conscience. p 199.
- Arcila, C; Balbosa, E; Cabezuelo, F. 2016. Técnicas BigData: Análisis de texto a gran escala para la investigación científica y periodística. Segovia, ES. Revista RECYT. Vol. 25. p 625.
- Bauer, J y Vega, R. 2015. Algoritmo que ejecuta submodelos de un modelo matemático para mayor eficiencia. Montecillo, MX. Revista Scielo. Vol. 16. p 551.
- Botero, A. 2015. La metodología documental en la investigación jurídica: alcances y perspectivas. Logroño. ES. Revista Dialnet. Vol. 2. p 111.
- ESPAM MFL (Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López). 2015a. Historia de ESPAM MFL. (En línea). EC. Consultado, 08 de may. 2017. Formato HTML. Disponible en: <http://web1.espam.edu.ec/?item=15>
- ESPAM MFL (Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López). 2015b. Misión y visión de la Carrera de Computación. (En línea). EC. Consultado, 08 de may. 2017. Formato HTML. Disponible en: <http://computacion.espam.edu.ec/>
- ESPAM MFL (Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López). 2017. Modelo Educativo. Calceta-Manabí, EC. p 34.
- Franco, E. 2013. Análisis de algoritmos. (En línea). MX. Consultado, 07 de may. 2017. Formato PDF. Disponible en: <http://www.eafranco.com/docencia/analisisdealgoritmos/files/02/Tema02.pdf>
- Garza, J. 2013. Aplicación para el diseño de experimentos para el análisis de secado de un producto. Monterrey.MX. Revista InnOvaciones de NegOcios UANL. p 146.

- Joyanes, L. 2013. Big Data, análisis de grandes volúmenes de datos en organizaciones. 1 ed. México.
- Lin, J y Dyer, C. 2010. Data-Intensive Text Processing with MapReduce.1 ed. Estados Unidos. p 1,2.
- Maya, E. 2014. Métodos y técnicas de investigación. 1 ed electrónica. México. p 13, 14.
- Manterola, C y Otzen, T. 2015. Estudios experimentales 2 Parte. Estudios Cuasi-Experimentales. Temuco. CL. Revista Scielo. Vol. 33. p 382.
- Moreno, E y Rubilar, F. 2015. Búsqueda en texto. (En línea). CL. Consultado, 03 de may. 2017. Formato PDF. Disponible en: https://www.u-cursos.cl/usuario/777719ab2ddbdb16d99df29431d3036/mi_blog/r/Tarea_1_Logaritmos.pdf
- Navarro, P; Ottone, NE; Acevedo, C; Cantín, M. 2017. Pruebas estadísticas utilizadas en revistas odontológicas de la Red Scielo. Temuco. CL. Revista Scielo. Vol. 33. p 30.
- Pérez, J y Lopera, I. 2016. Gestión humana de orientación analítica: un camino para la responsabilización. Medellín. CO. Revista Scielo. Vol. 56. p 104.
- Porre, J. 2013. La investigación bibliográfica. (En línea). CL. Consultado, 04 de feb. 2017. Formato HTML. Disponible en: <http://mtu-pnp.blogspot.com/2013/07/la-investigacion-bibliografica.html>
- Ramírez, K. 2014. Algoritmos de búsqueda secuencial de texto. (En línea). CR. Consultado, 09 de nov. 2016. Formato PDF. Disponible en: <http://www.kramirez.net/wp-content/uploads/2012/02/Algoritmos-de-Busqueda-Secuencial-de-Texto.pdf>
- Reutter, J. 2015. El modelo detrás de MapReduce. (En línea). CL. Consultado, 02 de may. 2017. Formato PDF. Disponible en: https://www.ing.uc.cl/ciencia-de-la-computacion/wp-content/uploads/2015/01/juan-reutter_mapreduce.pdf

- Rivera, T. 2013. Análisis de varianza de dos factores con varias muestras. (En línea). Consultado, 29 de jul. 2017. Formato HTML. Disponible en: https://prezi.com/c0cfqbkvbnh/analisis-de-varianza-de-dos-factores-con-varias-muestras/#_=_
- Rodríguez, A y Simón, A. 2013. Método para la extracción de información estructurada desde textos. La Habana, CU. Revista Scielo. Vol. 7. p 56.
- Rosa, V y Rivera, J. 2016. Big Data, análisis de datos en la nube. San Salvador. SV. Revista Entorno. p 19.
- Rubio, M y Berlanga, V. 2012. Cómo aplicar las pruebas paramétricas bivariadas t de Student y ANOVA en SPSS. Caso práctico. Barcelona, ES. Revista REIRE. Vol. 5. p 96.
- Sánchez, C; Torres, J; Betancourt, A. 2012. Búsqueda de texto. (En línea). Consultado, 09 de nov. 2016. Formato PDF. Disponible en: <https://prezi.com/vcvsgz7hmek3/busqueda-en-texto/>
- Torres, M; Páez, P; Martínez, C; Rodríguez, J. 2012. Algoritmos para la búsqueda de texto. Bogotá. CO. Revista International Institute of Informatics and Systemics. p 2.
- Torrado, M y Berlanga, V. 2013. Análisis discriminante mediante SPSS. Barcelona, ES. Revista REIRE. Vol. 6. p 166.

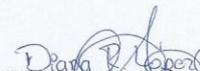
ANEXOS

Anexo 1. Acuerdo de algoritmos a analizar.**ESCUELA SUPERIOR POLITÉCNICA AGROPECUARIA DE
MANABÍ MANUEL FÉLIX LÓPEZ****CARRERA DE COMPUTACIÓN**

Calceta, 02 de mayo de 2017

Luego de buscar información sobre los algoritmos de búsqueda de patrones de texto, de acuerdo con el tutor y las postulantes se determina que los algoritmos enlistados son aquellos que se van a ser analizados y comparados para determinar su eficiencia, eficacia y así establecer cual o cuales presentan un mejor comportamiento.

Listado de algoritmos	
Nº	Algoritmos
1	Fuerza Bruta
2	Boyer-Moore Hospool
3	Knuth-Morris-Pratt
4	MapReduce


.....
Ing. Luis Cedeño
Tutor
.....
Carmen Cedeño
Postulante
.....
Diana López
Postulante

Anexo 2. Ficha de observación utilizada para el análisis de los algoritmos.

FICHA DE OBSERVACIÓN PARA EL ANÁLISIS DE LOS ALGORITMOS						
Objetivo: Analizar los algoritmos de búsqueda de patrones de texto					Fecha:	
Algoritmos			Indicador de eficacia	Indicadores de eficiencia		
Nombre	Referencia bibliográfica		Cantidad de aciertos	Cantidad de recursos que utiliza		Tiempo de respuesta
	Autor	Link		Para almacenar (RAM)	Para procesar (RAM)	
Fuerza Bruta	Sandova, Y. 2011	http://xcodigoinformatico.blogspot.com/2011/07/algoritmo-boyer-moore-string-matching.html	73 de 73	23,9 MB	0,5 MB	0,124 ms
Boyer-Moore-Horspool	Cruz, M. 2012	http://blog.martincruz.me/2012/09/algoritmos-de-busqueda-boyer-moore.html	7 de 77	23,7 MB	0,4 MB	0,030 ms
Knuth-Morris-Pratt	Fernández, J. 2014	http://codebotic.blogspot.com/2014/06/casamiento-de-cadenas-algoritmo-de.html	62 de 73	25,4 MB	0,5 MB	0,203 ms
MapReduce	Tutorials Point. 2016	https://www.tutorialspoint.com/es/hadoop/hadoop_mapreduce.htm	Se analizaron 1340811 filas y se encontraron 6 coincidencias		6,79 GB	19,3 ms

Anexo 3. Listado de indicadores de eficacia y eficiencia de los algoritmos.

Anexo 3-A. Algoritmo Fuerza Bruta.

LISTADO DE INDICADORES DE EFICACIA Y EFICIENCIA DEL ALGORITMO FUERZA BRUTA					
No.	NOMBRE DEL INDICADOR	TIPO DE INDICADOR	PARA QUE SIRVE EL INDICADOR	RESULTADO	RESPONSABLES
1	Cantidad de aciertos	EFICACIA	Cantidad de coincidencias encontradas en un texto	73 de 73	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
2	Recursos para almacenar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para almacenar	23,9 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
3	Recursos para procesar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para el procesamiento.	0,5 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
4	Tiempo de respuesta	EFICIENCIA	Medir el tiempo que emplea el algoritmo en mostrar el resultado	0,124 ms	Carmen Gema Cedeño Cool Diana Raquel López Cedeño

Anexo 3-B. Algoritmo Boyer-Moore-Horspool.

LISTADO DE INDICADORES DE EFICACIA Y EFICIENCIA DEL ALGORITMO BOYER-MOORE-HORSPPOOL					
No.	NOMBRE DEL INDICADOR	TIPO DE INDICADOR	PARA QUE SIRVE EL INDICADOR	RESULTADO	RESPONSABLES
1	Cantidad de aciertos	EFICACIA	Cantidad de coincidencias encontradas en un texto	7 de 77	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
2	Recursos para almacenar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para almacenar	23,7 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
3	Recursos para procesar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para el procesamiento.	0,4 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
4	Tiempo de respuesta	EFICIENCIA	Medir el tiempo que emplea el algoritmo en mostrar el resultado	0,030 ms	Carmen Gema Cedeño Cool Diana Raquel López Cedeño

Anexo 3-C. Algoritmo Knuth-Morris-Pratt.

LISTADO DE INDICADORES DE EFICACIA Y EFICIENCIA DEL ALGORITMO KNUTH-MORRIS-PRATT					
No .	NOMBRE DEL INDICADOR	TIPO DE INDICADOR	PARA QUE SIRVE EL INDICADOR	RESULTADO	RESPONSABLES
1	Cantidad de aciertos	EFICACIA	Cantidad de coincidencias encontradas en un texto	62 de 73	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
2	Recursos para almacenar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para almacenar	25,4 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
3	Recursos para procesar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para el procesamiento.	0,5 MB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
4	Tiempo de respuesta	EFICIENCIA	Medir el tiempo que emplea el algoritmo en mostrar el resultado	0,203 ms	Carmen Gema Cedeño Cool Diana Raquel López Cedeño

Anexo 3-D. Algoritmo MapReduce.

LISTADO DE INDICADORES DE EFICACIA Y EFICIENCIA DEL ALGORITMO MAPREDUCE					
No .	NOMBRE DEL INDICADOR	TIPO DE INDICADOR	PARA QUE SIRVE EL INDICADOR	RESULTADO	RESPONSABLES
1	Cantidad de aciertos	EFICACIA	Cantidad de coincidencias encontradas en un texto	Se analizaron 1340811 filas y se encontraron 6 coincidencias	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
3	Recursos para procesar (RAM)	EFICIENCIA	Para medir la cantidad de recurso (RAM) que el algoritmo emplea para el procesamiento.	6,79 GB	Carmen Gema Cedeño Cool Diana Raquel López Cedeño
4	Tiempo de respuesta	EFICIENCIA	Medir el tiempo que emplea el algoritmo en mostrar el resultado	19,3 s	Carmen Gema Cedeño Cool Diana Raquel López Cedeño