

**Institución: Instituto Tecnológico de Costa Rica**

**Área: Ingeniería en Computadores**

**Curso: Algoritmos y Estructuras de Datos I**

**Profesor: Antonio Gonzales Torres**

**Grupo: 02**

**Documento Anexo**

**Integrantes:**

- **Gustavo Alvarado Aburto | 2021055170.**
- **Byron Mata Fuentes | 2021430403.**

**Año: 2021**

## **Documento Anexo**

## Tabla de contenido

I. Introducción.....	1
II. Diseño.....	2
a) Listado de requerimientos.....	2
Requerimientos:.....	2
b) Estado del arte:.....	2
- Estructuras de datos:.....	2
- Arboles como estructura de datos: .....	3
- Arboles binario de expresiones: .....	3
- Notación Posfija: .....	3
- Interfaz web:.....	3
c) Estándares o normas: .....	3
Convenciones referentes al código.....	3
Convenciones referentes al documento .....	4
Convenciones referentes al diagrama .....	4
III. Trabajo individual y evaluación .....	4
a) Metas del proyecto:.....	4
b) Roles: .....	5
c) Reglas:.....	5
d) Cronograma: .....	6
e) Bitácora: .....	7
[9 de octubre (16:30 - 22:00)]: .....	7
[16 de octubre (15:30 – 19:00)]: .....	7
[19 de octubre (17:00 – 22:30)]: .....	8
[25 de octubre ( 17:30 - 23:00)]: .....	8
[26 de octubre (16:20 - 22:40)]: .....	9
[28 de octubre ( 18:30 - 19:40)]: .....	9
IV. Referencias .....	9

## *I. Introducción*

Este escrito es un documento anexo de lo que es el proyecto II del curso de algoritmos y estructuras de datos I, el mismo tiene la finalidad de “Elaborar un documento que evidencie la capacidad para la describir un problema complejo de ingeniería en términos de requerimientos de diseño, limitantes y la capacidad para la organizar las actividades individuales en el equipo de trabajo”; así mismo también se puede incluir: “Determinar los requerimientos de ingeniería para un problema complejo considerando partes involucradas, estado del arte, estándares, normas, entre otras. Establecer los lineamientos para el trabajo en equipo (metas, roles, reglas, cronogramas, bitácoras, entre otros). Identificar el rol y las responsabilidades como miembro de un equipo de trabajo”, con los anteriores objetivos se pretende poner en acción los atributos de Diseño (Inicial) y el de Trabajo individual y en equipo (Inicial). Son los anteriores aspectos y/o apartados mencionados en el párrafo anterior los que conforman el cuerpo de este documento, únicamente adicionar que se le agrega un apartado donde se encontraran algunas de las referencias o fuentes bibliográficas utilizadas para el apartado del Estado de Arte.

## *II. Diseño*

### a) Listado de requerimientos.

#### Requerimientos:

1. Arquitectura cliente/servidor utilizando sockets TCP que soporte n número de clientes.
2. Interfaz web sencilla que permite el ingreso de expresiones matemáticas de cualquier longitud que usan los operadores +, -, \*, /, %, ( y ).
3. Ingreso de expresiones desde el cliente.
4. Envío de operaciones del cliente al servidor.
5. Evaluación de expresiones en el servidor y envío de resultado devuelta al cliente.
6. Despliegue de resultado en un recuadro para el cliente.
7. Registro de operaciones realizadas por cada cliente en el servidor, debe incluir: fecha, expresión y resultado en un archivo csv.
8. Consulta del registro de expresiones por parte del cliente.
9. Evaluación de distintas expresiones al mismo tiempo en el servidor.

### b) Estado del arte:

#### - Estructuras de datos:

Según comenta Roger Edgar Yarleque Ferrer “Una estructura de datos es una colección de datos que pueden ser caracterizados por su organización y las operaciones que se definen en ella”(2018). Por otro lado, Joel Ayala de la Vega (2017), define este concepto de la siguiente manera “una estructura de datos es una forma particular de organizar datos en una computadora para que pueda ser utilizado de manera eficiente”.

#### -Arboles como estructura de datos:

Roger Edgar Yarleque Ferrer (2018), comenta acerca de este concepto que: “Un árbol es una estructura no lineal en la que cada nodo puede apuntar a uno o varios nodos” a la vez menciona que a este concepto se le suele definir de forma recursiva como: “un árbol es una estructura en compuesta por un dato y varios árboles”. Otra definición no muy alejada de las anteriores indica que “Los árboles son las estructuras de datos no lineales y dinámicas de datos más importante del área de computación” (Cairó & Guardati, 2006, pp. 177).

#### - Arboles binario de expresiones:

De acuerdo con Isac Méndez (2010), los árboles de expresiones son aquellos que representan el código de nivel de lenguaje en forma de datos, estos se almacenan en una estructura con forma de árbol y cada nodo de este representa una expresión, por ejemplo, una llamada al método o una operación binaria. José M. Badía, Begoña Martínez, Antonio Morales y José M. Badía (2012) señalan que “Un árbol binario de expresiones es la representación de una expresión aritmética mediante un árbol binario”.

#### - Notación Posfija:

Alexander Borbón lo define como “forma de escribir una expresión matemática tiene la ventaja que se evalúa de forma lineal por lo que es más sencillo para una computadora entenderlo”(2006), a la anterior definición se le puede adjuntar la dada por Estaban Gómez (2016), que menciona que es una variación de la notación prefija de forma que el operador se pone después que los operandos.

#### - Interfaz web:

“Se denomina interfaz al conjunto de elementos de la pantalla que permiten al usuario realizar acciones sobre el Sitio Web que está visitando”(Guía Digital Beta, 2008).

#### c) Estándares o normas:

#### Convenciones referentes al código:

1. Los package se escribirán en minúscula a excepción de los que se crean al realizar la importación de alguna tecnología o bien una librería.
2. Las clases del proyecto se escriben en mayúscula.
3. Los constructores de las clases se escribirán en mayúscula y deben de ir acorde al nombre de la clase.
4. Los nombres de métodos de una clase conformados por una palabra se escribirán en minúscula.
5. Los nombres de métodos de una clase conformados por dos palabras la primera de estas se mantiene en minúscula y la segunda únicamente la inicial se escribirá en mayúscula.
6. El nombre de los métodos y de las variables a utilizar deberán tener lógica o relación con la funcionalidad que cumplan o tengan en sí.
7. En los métodos ante la presencia de condicionales estos se agrupan en bloques, es decir, se realiza una separación interna de una línea de código.
8. Para la documentación interna del código, se utilizará el formato de encabezado que se ofrece para el JavaDoc es decir “/\*\*, \*/”.
9. Los comentarios por línea para la documentación se realizarán en lugares estratégicos para que se logre contextualizar de manera algo general que sucede en el lugar ubicado.
10. Para la documentación interna o JavaDoc no se hizo uso de tildes en las palabras para evitar errores en la elaboración de este.

#### Convenciones referentes al documento:

1. El documento anexo, así como la wiki del repositorio se elaborará bajo normativa formato APA 7ª edición.

#### Convenciones referentes al diagrama:

1. El diagrama se elaborará siguiendo la guía del lenguaje UML.

### ***III. Trabajo individual y evaluación***

#### **a) Metas del proyecto:**

- 1. Entregar el proyecto en su completitud.**

2. Desarrollar la aplicación o bien el proyecto cumpliendo los requerimientos solicitados en el enunciado.
3. Entregar en su completitud los documentos de la documentación externa e interna, el proyecto en Azure DevOps y el anexo del proyecto.
4. Aplicar y cumplir las diferentes reglas establecidas para el desarrollo del proyecto.
5. Cumplir en la medida de lo posible con el cronograma planteado.

b) Roles:

Roles	Nombre
Líder	<ul style="list-style-type: none"> <li>• Gustavo Alvarado Aburto</li> </ul>
Desarrollador(es)	<ul style="list-style-type: none"> <li>• Byron Mata Fuentes</li> <li>• Gustavo Alvarado Aburto</li> </ul>
Investigador(es)	<ul style="list-style-type: none"> <li>• Gustavo Alvarado Aburto</li> <li>• Byron Mata Fuentes</li> </ul>

c) Reglas:

1. Respeto bajo cualquier concepto a los integrantes del grupo de trabajo.
2. Promover una comunicación asertiva entre los miembros del grupo de trabajo.
3. Compromiso hacia, para y con el proyecto e integrantes del grupo de trabajo.
4. Puntualidad con las horas asignadas, para las tareas y reuniones.
5. Realizar aportes en la solución de problemas, desafíos, entre otros.
6. Mostrar iniciativa en el desarrollo del proyecto, por medio de la participación.
7. Realizar una retroalimentación a los aportes realizados por los miembros, en caso de ser necesario.
8. Mantener informados a los miembros del grupo de trabajo de cualquier actualización realizada.
9. Luego de cualquier actualización, indicar lo realizado en la bitácora, de igual manera con alguna conclusión o recomendación.
10. Trabajar de manera clara y ordenada.



d) Cronograma:

<b>Actividad</b>	<b>Asignada a...</b>	<b>Día de asignación</b>	<b>Día de entrega</b>
Conexión servidor de aplicaciones (Java EE)	Byron Mata Gustavo Alvarado	9 de octubre	19 de octubre
Elaboración de anexo	Byron Mata	11 de octubre	29 de octubre
Elaboración de wiki	Byron Mata	11 de octubre	29 de octubre
Elaboración del proyecto en Azure DevOps	Gustavo Alvarado	11 de octubre	17 de octubre
Creación del repositorio en GitHub	Gustavo Alvarado	19 de octubre	20 de octubre
Interfaz web	Gustavo Alvarado	19 de octubre	20 de octubre
Stack & Queue	Byron Mata	19 de octubre	20 de octubre
Captura y envío de datos	Byron Mata	24 de octubre	25 octubre
Árbol de expresiones	Gustavo Alvarado Byron Mata	20 de octubre	27 de octubre
CSV	Gustavo Alvarado	25 de octubre	27 de octubre
Diagrama de clases	Byron Mata Gustavo Alvarado	27 de octubre	27 de octubre
JavaDoc	Byron Mata Gustavo Alvarado	---	---
Entrega del proyecto	Byron Mata Gustavo Alvarado	---	29 de octubre

e) Bitácora:

[9 de octubre (16:30 - 22:00)]:

La reunión se llevó a cabo en una llamada de Discord donde se realizó en conjunto las siguientes tareas:

- 1- Se realizó una investigación relacionada a la implementación de los JSF y de Primefaces en el IDEA.
- 2- Se definió la configuración del IDEA para el desarrollo del proyecto.
- 3- Se logró aprender en pequeña medida como realizar una interfaz con Primefaces, es decir, como se implementan los componentes que ofrece.
- 4- Se elaboró una interfaz sencilla con JSF y Primefaces, esta fue pensada como la posible opción definitiva.

**Byron:** Luego de realizar la investigación se consiguió realizar la implementación y configuración del IDEA en el apartado de la implementación de Apache Tomcat para el servidor, así mismo la implementación de la dependencia correspondiente a Primefaces, además logró ejecutar el código y mostrar en el navegador los componentes de Primefaces.

**Gustavo:** Previo a iniciar la investigación, definió la configuración para el IDEA, es decir, decidió el SDK o bien el JDK a utilizar para el desarrollo, así mismo decidió el uso de Tomcat sobre GlassFish puesto que estas eran las dos opciones para los servidores, lo mismo con el uso de la librería de Primefaces, finalmente desarrollo la interfaz simple mencionada al inicio.

[16 de octubre (15:30 – 19:00)]:

Mediante una llamada en Discord se empezaron los a crear los documentos escritos referentes a la documentación externa, así como, el proyecto en Azure DevOps y el anexo del proyecto.

**Byron:** Se encargó de empezar a elaborar el documento anexo del proyecto, creando el espacio de cada apartado que indicaba el enunciado para seguidamente empezar a rellenar o redactar lo que en estos se le solicitaba, por otro lado, con la información que poseía empezó a elaborar un borrador de lo que sería la Wiki del repositorio.

**Gustavo:** Se encargó de elaborar el listado de las historias de usuario, así como de crear el proyecto en Azure DevOps, donde estas irían incluidas.

[19 de octubre (17:00 – 22:30)]:

Mediante una llamada en Discord se crearon las clases de java para la pila y cola que se utilizan al crear la expresión en notación posfija, además se crearon las clases para el árbol de expresión binario y sus nodos. También, utilizó html y css para crear y añadir los componentes gráficos a la página web.

**Byron:** Se dedico a la creación de varias clases que contienen algunos métodos que se implementarían en la pila y en la cola, estos métodos abarcan los setter y getter entre algunos otros. Finalmente, empezó a realizar la documentación interna de las clases, constructores y métodos de las clases mencionadas, esto para realizar una mejor inversión del tiempo.

**Gustavo:** Se encargó de añadir y crear los componentes gráficos a la página web, así como la creación de las clases para el árbol de expresión binario con sus respectivos métodos (getter y setter para la clase nodo, método para notación posfija, y constructores) y el repositorio de GitHub con el proyecto cargado. Finalmente se realizaron pruebas para comprobar en funcionamiento de la notación posfija y el código funcionó correctamente.

[25 de octubre ( 17:30 - 23:00)]:

De forma individual se creó el archivo JSP para realizar la captura de datos ingresados en la página web, además se añadió el método para crear el árbol de expresión binario a su respectiva clase para crear el árbol de expresión.

**Byron:** Implementar la captura de datos ingresados en la etiqueta `<input type="text">` de la interfaz web para enviarlos hasta el archivo .jsp, para seguidamente llamar el método para el procesado de la expresión, es decir, `organizeQS()`, sin embargo, la implementación realizada no resulto útil para el resultado que se esperaba puesto que pese a ser funcional, se presentaban bugs de interacción con los elementos de la aplicación web.

**Gustavo:** Se encargó de realizar el método para crear el árbol de expresión binario con base a la cola que almacena la notación posfija, además añadió los métodos getter y setter del identificador del nodo a la clase que contiene los nodos del árbol de expresión.

[26 de octubre (16:20 - 22:40)]:

**Byron:** Implementó en la interfaz web un tipo de Modal PopUp donde se muestra el resultado o la evaluación del árbol de expresiones, para ello se utilizó el css y para la funcionalidad de aparecer y desaparecer al presionar un botón se implementó lenguaje referente a JavaScript. Se realizó otra implementación para el almacenamiento de datos donde estos se enviaban por medio de un formulario hasta una clase java, la cual extendía a HttpServlet y estos se reenviaban hasta el archivo .jsp, sin embargo, esto reiniciaba a la aplicación de la página lanzada, por lo que no se logró obtener el resultado esperado. Finalmente, empezó la elaboración de la Wiki del repositorio para concluir con la misma y únicamente esperar a la elaboración del diagrama de clases para agregarlo a la Wiki.

**Gustavo:** Se encargó de realizar el método para la evaluación del árbol de expresión binario el cual realizar la evaluación partiendo del nodo raíz del árbol, luego recorre recursivamente, además de crear los métodos para el guardado de datos en un archivo .csv en el cual se mantiene un registro de la expresión ingresada, fecha y resultado de la expresión.

[28 de octubre ( 18:30 - 19:40)]:

Se realizó en ultimo commit del proyecto, también se terminó la documentación interna, así como la wiki del repositorio, además se realizó el Javadoc y el diagrama de clases para posteriormente proceder a realizar una revisión global del proyecto siguiendo los detalles especificados en el enunciado de este para finalmente dar luz verde a la conclusión total del proyecto, seguido de ello se realizó la entrega de este en TecDigital.

#### *IV. Referencias*

- Ayala, J. (2017). Ingeniería en computación. Apuntes. Estructuras de datos. Universidad Autónoma del Estado de México.  
[http://ri.uaemex.mx/bitstream/handle/20.500.11799/69935/secme-17408\\_1.pdf](http://ri.uaemex.mx/bitstream/handle/20.500.11799/69935/secme-17408_1.pdf)
- Badía, J., Martínez Salvador, B., Morales Escrig, A., & Sanchiz Martí, J. M. (2012). Tema 10. Árboles [PDF].  
<http://repositori.uji.es/xmlui/bitstream/handle/10234/119889/tema10.pdf;jsession>
- Cairó, O., & Guardati, S. (2006). *Estructuras de datos*. MC GRAW HILL INTERAMERICANA.  
<http://up-rid2.up.ac.pa:8080/xmlui/handle/123456789/1299>

- Gomez, E. (2016). Estructura de Datos y de la Información. Pilas y expresiones aritméticas. Laboratorio de Investigación y desarrollo en Inteligencia Artificial. Departamento de Computación. Universidad de la Coruña, España.  
<https://docs.google.com/viewer?a=v&pid=sites&srcid=ZXNwZS5lZHUuZWw4cHJvZ3JhbWVjaW9uLWlpcGd4OjIzMTlhODYxM2E1YWUzYjE>
- Guía Digital. (2008). ¿Qué es una interfaz web? Guía Digital.  
<https://www.guiadigital.gob.cl/articulo/que-es-una-interfaz.html#:~:text=Cuando%20se%20habla%20de%20Sitios,de%20contenidos%20y%20de%20acci%C3%B3n>.
- Mendez, I. (2010). Árbol de Expresión. Isc. Mendez- Portafolio.  
<https://sites.google.com/site/iscmendezportafolio/arboles-de-expresion>
- Yarleque Ferrer, R. E. (2018). Estructura de datos Introducción [PDF].  
<https://repositorio.une.edu.pe/handle/UNE/3109>