



# ESCUELA POLITÉCNICA NACIONAL

## ESCUELA DE FORMACIÓN DE TECNÓLOGOS



### ALGORITMOS Y ESTRUCTURAS DE DATOS

ASIGNATURA: Algoritmos y Estructuras de Datos  
PROFESOR: Ing. Loarte Byron  
PERÍODO ACADÉMICO: 2020A

### DEBER 21

### TÍTULO:

## Búsqueda Binaria y Exhaustiva

### ESTUDIANTE

Cindy Kathalina Yazán Chuquimarca

FECHA DE REALIZACIÓN: 17/08/2020

FECHA DE ENTREGA: 18/08/2020

CALIFICACIÓN OBTENIDA:

FIRMA DEL PROFESOR:

# 1 PROPÓSITO DE LA PRÁCTICA

Realizar los ejercicios planteados en la práctica.

## 2 OBJETIVOS ESPECÍFICOS

- Determinar las ventajas y desventajas de la Búsqueda binaria.
- Realizar los recorridos de búsqueda en profundidad y en anchura.
- Elaborar un análisis de la Búsqueda Exhaustiva.

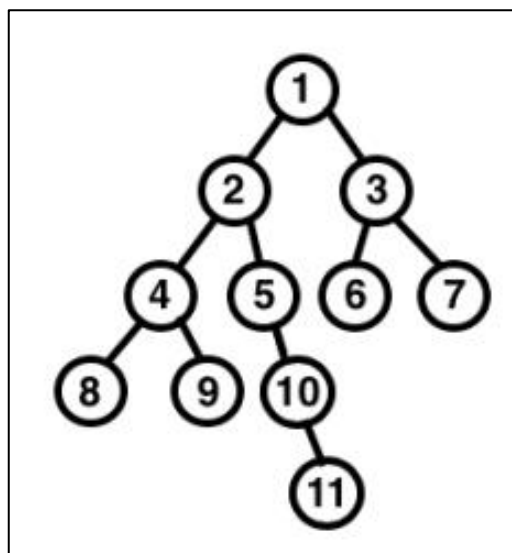
## 3 DESARROLLO Y RESULTADOS DE LA PRÁCTICA

1. Se realiza un análisis del algoritmo de Búsqueda binaria

*Tabla 1. Ventajas y Desventajas de la Búsqueda binaria [1]*

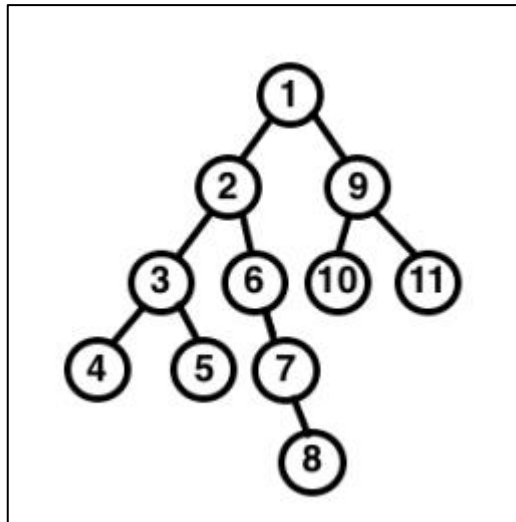
Algoritmo de Búsqueda Lineal.	
Ventajas	Desventajas
<ul style="list-style-type: none"> <li>• Su funcionamiento es óptimo con listas o arreglos de todo tamaño.</li> <li>• Utiliza la mitad del arreglo porque usa el método de “divide y vencerás”.</li> <li>• En el mejor de los casos el dato a buscar se encuentra en la mitad.</li> <li>• Es sencilla la implementación del algoritmo.</li> </ul>	<ul style="list-style-type: none"> <li>• Es necesario tener el arreglo o lista ordenados.</li> <li>• Si el arreglo no está ordenado el algoritmo no sirve, se realiza el algoritmo pero no de la forma correcta.</li> </ul>

2. Se realiza el DFS y BFS para cada árbol.



DFS = {1, 2, 4, 8, 9, 5, 10, 11, 3, 6, 7}

BFS = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11}



DFS = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

BFS = {1, 2, 9, 3, 6, 10, 11, 4, 5, 7, 8}

Posterior al análisis se encuentra el análisis del algoritmo de Búsqueda Exhaustiva.

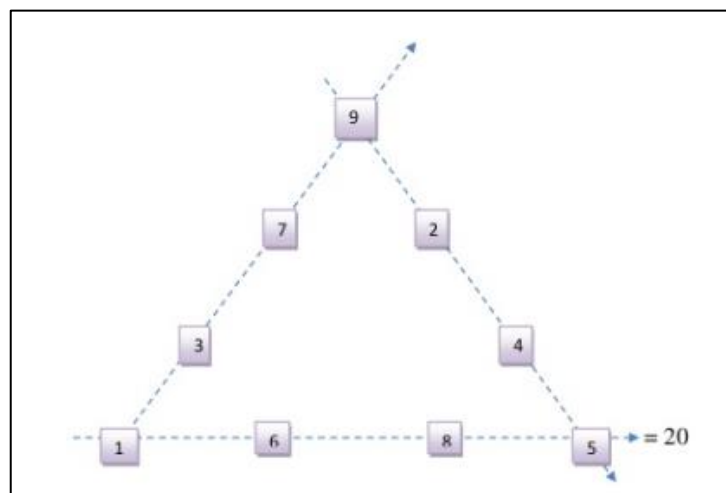
Esta búsqueda trata de un tipo de algoritmo de fuerza bruta, que se lo utiliza en problemas donde se busca un elemento con una cualidad especial, comúnmente objetos como permutaciones, combinaciones o subconjuntos [2].

Consiste en:

1. Elaborar una lista de todas las posibles soluciones que existan.
2. Evaluar las potenciales soluciones, erradicar las que no sean factibles.
3. Mantener un registro de las mejores soluciones y elegir la mejor al finalizar.

Un ejemplo de esta búsqueda es el siguiente:

Coloca los dígitos del 1 al 9 en los cuadros de la figura de abajo, de forma que la suma de la suma de los cuatro números que forman cada lado sume 20 [2].



Respuestas posibles:

1 3 7 9

1 4 6 9

1 5 6 8

2 4 5 9

2 4 6 9

2 5 6 7

3 4 5 8

3 4 7 6

Un algoritmo de búsqueda exhaustiva es debidamente rápido, solo en instancias pequeñas como en caminos cortos, arboles de expansión mínima entre otras.

En varios casos, la búsqueda exhaustiva es la mejor solución para buscar de forma exacta [3].

Se puede encontrar esta información en la siguiente dirección:

## 4 CONCLUSIONES

Se han determinado las ventajas y desventajas de la búsqueda binaria y se ha concluido que es la búsqueda más útil en datos de cualquier rango de tamaño. Además se han realizado las dos búsquedas en ancho y profundidad de dos árboles y por último se ha desarrollado el análisis del algoritmo de búsqueda exhaustiva con un ejemplo práctico y se todo se puede encontrar en un repositorio de GitHub.

## 5 BIBLIOGRAFÍA

- [1] «ALGORITMOS Y ESTRUCTURAS DE DATOS».
- [2] SlideShare, «PROBLEMAS DE BUSQUEDA EXHAUSTIVA. EJERCICIOS DE CONSOLIDACION». [En línea]. Disponible en: <https://es.slideshare.net/lissethkatt/problemas-de-busqueda-exhaustiva-ejercicios-de-consolidacion>. [Accedido: 17-ago-2020].
- [3] E. A. R. Tello, «Algoritmos de búsqueda exhaustiva».