

Algoritmo de ordenación no recursivo heapsort

Patricio Aguirre, Ricardo Ruales, Diego Sotomayor

**Sistemas Multimedia**

**Proyecto Final**

**Avance 1**

**Nombres:** Patricio Aguirre, Ricardo Ruales, Diego Sotomayor

**Algoritmo de ordenación no recursivo Heapsort**

**Objetivos**

**General:**

* Diseñar una página web con los recursos aprendidos, explicando e implementando el algoritmo de ordenación HeapSort

**Específicos:**

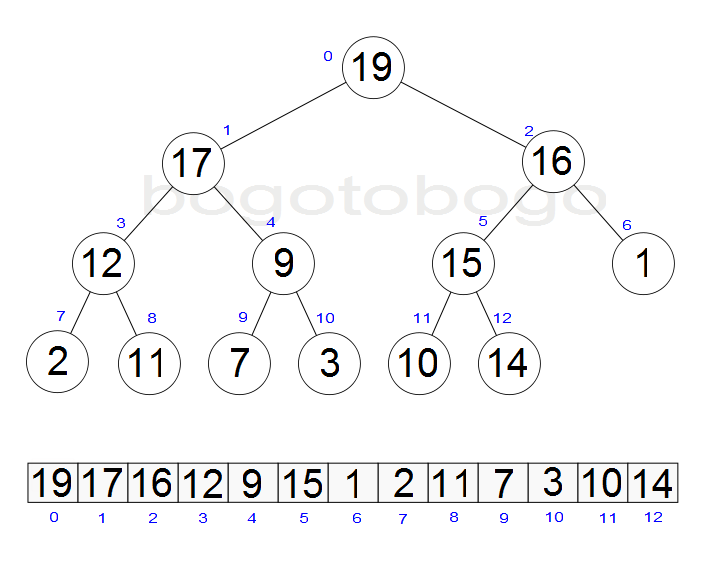
* Se diseñará la página web con una interfaz amigable y con explicación del algoritmo seleccionado
* Se usará HTML, CSS y JavaScript para diseñar las animaciones y el diseño que cumplan con la funcionalidad del algoritmo
* Se especificará datos, funcionalidad, propiedades, definición, ventajas y desventajas del tema seleccionado.

**Alcance:**

Esto se realiza con la finalidad de demostrar todo el conocimiento adquirido durante el periodo de aprendizaje, además de la investigación adicional impuesta por cada uno de los integrantes. Sin más motivar a las siguientes generaciones a diseñar páginas web con animaciones de algoritmos aprendidos con anterioridad y así alcanzar el objetivo común de la clase.

**Introducción**

El ordenamiento por montículos (heapsort en inglés) es un algoritmo de ordenamiento no recursivo, no estable, con complejidad computacional.



Este algoritmo consiste en almacenar todos los elementos del vector a ordenar en un montículo (heap), y luego extraer el nodo que queda como nodo raíz del montículo (cima) en sucesivas iteraciones obteniendo el conjunto ordenado. Basa su funcionamiento en una propiedad de los montículos, por la cual, la cima contiene siempre el menor elemento (o el mayor, según se haya definido el montículo) de todos los almacenados en él. El algoritmo, después de cada extracción, recoloca en el nodo raíz o cima, la última hoja por la derecha del último nivel. Lo cual destruye la propiedad heap del árbol. Pero, a continuación, realiza un proceso de "descenso" del número insertado de forma que se elige a cada movimiento el mayor de sus dos hijos, con el que se intercambia. Este intercambio, realizado sucesivamente "hunde" el nodo en el árbol restaurando la propiedad montículo del árbol y dejando paso a la siguiente extracción del nodo raíz.

**Bases Conceptuales**

El algoritmo, en su **implementación** habitual, tiene dos fases.

* Primero una fase de construcción de un montículo a partir del conjunto de elementos de entrada, y después, una fase de extracción sucesiva de la cima del montículo. La implementación del almacén de datos en el heap, pese a ser conceptualmente un árbol, puede realizarse en un vector de forma fácil. Cada nodo tiene dos hijos y por tanto, un nodo situado en la posición i del vector, tendrá a sus hijos en las posiciones 2 x i, y 2 x i +1 suponiendo que el primer elemento del vector tiene un índice = 1. Es decir, la cima ocupa la posición inicial del vector y sus dos hijos la posición segunda y tercera, y así, sucesivamente.
* Por tanto, en la fase de ordenación, el intercambio ocurre entre el primer elemento del vector (la raíz o cima del árbol, que es el mayor elemento de este) y el último elemento del vector que es la hoja más a la derecha en el último nivel. El árbol pierde una hoja y por tanto reduce su tamaño en un elemento. El vector definitivo y ordenado, empieza a construirse por el final y termina por el principio.

**Descripción**

Se pueden encontrar descripciones de las operaciones insertar\_en\_monticulo y extraer\_cima\_del\_monticulo en el artículo sobre montículos.

* **function** heapsort(**array** A[0..n]):
* **montículo** M
* **integer** i; // declaro variable i
* **for** i = 0..n:
* **insertar\_en\_monticulo**(M, A[i])
* **for** i = 0..n:
* A[i] = **extraer\_cima\_del\_monticulo**(M)
* **return** A

**Desarrollo de Contenido**

**Tema principal**

Método de ordenación Heapsort

**Introducción:**

Los métodos de ordenamiento son ampliamente usados en el desarrollo de software, debido a que posibilitan tomar un grupo de datos y ordenarlos de manera secuencial en un tipo de datos de agrupación o arreglo.

**Tema 1: Conceptos Generales**

* 1. **Árbol binario**

Conjunto finito de nodos el cual puede ser vacío o tener un par de árboles llamados izquierdo y derecho.

Cuando un nodo no tiene hijos se le llama hoja o nodo terminal.

**1.2 Heap o montículo**

**1.2.1 Condiciones de árbol binario**

* El valor de cada nodo es mayor o igual a los valores almacenados en cada uno de sus hijos.
* El árbol está perfectamente balanceado y las hojas del ultimo nivel están todas en las posiciones en el extremo izquierdo.

**1.3 Funcionamiento**

Este algoritmo consiste en almacenar todos los elementos en un montículo y luego extraer el nodo que queda como raíz en iteraciones sucesivas obteniendo el conjunto ordenado.

**1.4 Conclusiones y referencias**

Concluyendo este método es conveniente cuando se trata de ordenar arreglos estáticos grandes a diferencia de otros métodos con el Quicksort y el Mergesort. Heapsort compite primariamente con Quicksort, otro método de ordenamiento muy eficiente para propósitos en general.

**Tema 2: Algoritmo HeapSort**

**2.1 Definición**

Este método es una variante del método por selección, donde la búsqueda del elemento máximo de un arreglo se realiza mediante técnicas basadas en la construcción de un montículo.

Puede ser un montículo ascendente o descendente.

Es un método de ordenamiento basado con comparación, usa el Montículo o Heap como estructura de datos. Este método es más lento que otros métodos, pero es más eficaz en escenarios más rigurosos.

Se define como un método No recursivo, No Estable y con Complejidad Computacional.

**2.2 Propiedades**

* Fue el primer algoritmo que surgió de los que ordenan en O(n log n).
* Ordena en el lugar. No utiliza memoria auxiliar
* No es estable.

**2.3 Montículo**

Es una estructura de datos del tipo árbol binario. Este árbol binario tiene que ser completo, en otras palabras, cada nivel debe de estar lleno con excepción del ultimo nivel, en el último nivel, los hijos deben recargarse hacia un mismo lado, generalmente hacia el lado izquierdo, así como se muestra en la imagen de la derecha.

**2.4 Aplicaciones**

Una de las más grandes aplicaciones de Heapsort es construir colas de prioridad con la idea que busqueda los procesos que llevan la mayor carga de prioridad dado una gran cantidad de procesos por hacer.

En esencia las aplicaciones de Heapsort son las que traten de sobre ordenar una lista de elementos.

**2.4.1 Ventajas**

Usar este método de Ordenamiento trae consigo diversas ventajas con respecto a los otros métodos, dichas características están en la tabla a continuación:

* Su desempeño es en promedio tan bueno como el Quicksort y se comporta mejor que este último en los peores casos.
* Funciona efectivamente con datos desordenados.
* No utiliza memoria adicional.

**2.4.2 Desventajas**

Usar este método de Ordenamiento trae consigo diversas desventajas con respecto a los otros métodos, dichas características están en la tabla a continuación:

* Aunque el Heapsort tiene un mejor desempeño general que cualquier otro método presentado de clasificación interna, es bastante complejo de programar.
* No es estable. Es un método complejo

**2.5 Contenido adicional**

**2.5.1 Algoritmos y estructuras de datos**

El algoritmo Heapsort pertenece a la familia de algoritmos de clasificación eficientes, cuya complejidad promedio es lo mejor que podemos encontrar. (Video adicional)

**2.5.2 ¿Cómo se construye?**

El método de Heapsort es un algoritmo de ordenamiento también llamado ordenamiento por montículos. Esta presentación tiene la explicación de cómo funciona el método utilizando montículos máximos. (Video adicional)

**2.5.3 Técnica de clasificación HeapSort**

El algoritmo de clasificación Heap tiene usos limitados porque Quicksort y Mergesort son mejores en la práctica. Sin embargo, la estructura de datos Heap en sí misma es enormemente utilizada. (Video adicional)

**Tema 3: Cuestionario**

**3.1 Preguntas importantes y/o frecuentes**

**¿Que son los métodos de ordenamiento?**

Un algoritmo de ordenamiento es un algoritmo que pone elementos de una lista o un vector en una secuencia dada por una relación de orden, es decir, el resultado de salida ha de ser una permutación o reordenamiento de la entrada que satisfaga la relación de orden dada.

**¿Qué es el método de ordenamiento Heapsort?**

Basa su funcionamiento en una propiedad de los montículos, por la cual, la cima contiene siempre el menor elemento (o el mayor, según se haya definido el montículo) de todos los almacenados en él. El algoritmo, después de cada extracción, recoloca en el nodo raíz o cima, la última hoja por la derecha del último nivel.

**¿Cuáles métodos de ordenamiento existen?**

Bubblesort,Cocktail sort,Insertion sort son los más comunes 4.

**¿Cuál es el método de ordenamiento más eficiente?**

Quicksort

**¿Qué condiciones debe cumplir el árbol de ordenamiento?**

Árbol binario que cumple ciertas condiciones:

1. El valor de cada nodo es mayor o igual a los valores almacenados en cada uno de sus hijos.

2. El árbol está perfectamente balanceado y las hojas del ultimo nivel están todas en las posiciones en el extremo izquierdo.

**¿Cuál es una aplicación del algoritmo heapsort?**

Una de las más grandes aplicaciones de Heapsort es construir colas de prioridad con la idea que búsqueda los procesos que llevan la mayor carga de prioridad dado una gran cantidad de procesos por hacer.

**Una Ventaja del algoritmo heapsort**

Su desempeño es en promedio tan bueno como el Quicksort y se comporta mejor que este último en los peores casos.

**Métodos/Procedimiento**

Para la primera fase del proyecto se crearon dos páginas web:

La página de inicio (index.html), aquí se puso una introducción al método de ordenamiento.

<!DOCTYPE HTML>

<!--

    Strongly Typed by HTML5 UP

    html5up.net | @ajlkn

    Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)

-->

<html>

    <head>

        <title>ALGORITMOS</title>

        <meta charset="utf-8" />

        <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />

        <link rel="stylesheet" href="assets/css/main.css" />

    </head>

    <body class="homepage is-preload">

        <div id="page-wrapper">

            <!-- Header -->

                <section id="header">

                    <div class="container">

                        <!-- Logo --><br><br><br><br>

                            </nav>

                            </nav>

                            <h1 id="logo"><a href="index.html">METODO DE ORDENACION<br><br>

                                                                HEAP SORT</a></h1>

                            <!-- Nav -->

                            <nav id="nav">

                                <ul>

                                    <li><a style="color:aliceblue" class="icon fa-home" href="index.html"><span>INTRODUCCION</span></a></li>

                                    <LI><a style="color:aliceblue" class="icon fa-cog" href="left-sidebar.html"><span>HEAP SORT</span></a></li>

                                        <li><a style="color:aliceblue" class="icon fa-question-circle"

                                            href="cuestionario.html"><span>Cuestionario</span></a></li>

                                </ul>

                                <img src="images/logo.png" width="700" height="200">

                    </div>

                </section>

            <!-- Features -->

                <section id="features">

                    <div class="container">

                        <header>

                            <h2 style="color:#000">Metodo de Ordenación por monticulos</h2>

                        </header>

                        <div class="row aln-center">

                        </div>

                    </div>

                </section>

            <!-- Banner -->

                <section id="banner">

                    <div class="container">

                        <p><strong>METODOS DE ORDENAMENTO</strong></p>

                        <p>Los métodos de ordenamiento son ampliamente usados en el desarrollo de software, debido a que posibilitan tomar un grupo de datos y ordenarlos de manera secuencial en un tipo de datos de agrupación o arreglo.</p>

                    </div>

                </section>

            <!-- Main -->

                <section id="main">

                    <div class="container">

                        <div class="row">

                            <!-- Content -->

                                <div id="content" class="col-8 col-12-medium">

                                        <article class="box post">

                                            <header>

                                                <h2 style="font-size:40px; color:#90C"><a ><strong>CONCEPTOS GENERALES</strong></a></h2>

                                            </header>

                                            <a href="#" class="image featured"><img src="images/algorithm-010.png" height="400" alt="" /></a>

                                            <h3 style="color:#00C">ARBOL BINARIO:</h3>

                                            <p  style="color:#000">Conjunto finito de nodos el cual puede ser vacío o tener un par de árboles llamados izquierdo y derecho.<br> Cuando un nodo no tiene hijos se le llama hoja o nodo terminal.</p>

                                            <h3 style="color:#00C">HEAP O MONTICULO:</h3>

                                            <p style="color:#000"><strong style="color:#000"> Árbol binario que cumple ciertas condiciones:</strong><br><br>

                                                <strong>1. </strong> El valor de cada nodo es mayor o igual a los valores almacenados en cada uno de sus hijos.<br><br>

                                                <strong>2. </strong> El árbol esta perfectamente balanceado y las hojas del ultimo nivel están todas en las<br>

                                                    posiciones en el extremo izquierdo.</p>

                                            <center><a href="#" class="#"><img src="images/ANI.gif" height="250" width="350" alt="" /></a></center>

                                            <ul class="actions">

                                                <li><a href="left-sidebar.html" class="button icon fa-file">CONTINUAR LEYENDO</a></li>

                                            </ul>

                                        </article>

                                </div>

                            <!-- Sidebar -->

                                <div id="sidebar" class="col-4 col-12-medium">

                                    <!-- Excerpts -->

                                        <section>

                                            <ul class="divided">

                                            </ul>

                                        </section>

                                    <!-- Highlights -->

                                        <section>

                                            <ul class="divided">

                                                <li>

                                                    <!-- Highlight -->

                                                        <article class="box highlight">

                                                            <header>

                                                                <h3 style="color:#00C; font-size:30px"><a href="#">FUNCIONAMIENTO</a></h3>

                                                            </header>

                                                            <a href="#" class="image left"><img src="images/440px-Complete\_binary.svg.png" alt="" /></a>

                                                            <p style="color:#000">Este algoritmo consiste en almacenar todos los elementos en un montículo y luego extraer el nodo que queda como raíz en iteraciones sucesivas obteniendo el conjunto ordenado.</p>

                                                            <ul class="actions">

                                                                <li><a href="right-sidebar.html" class="button icon fa-file">VER MAS</a></li>

                                                            </ul>

                                                        </article>

                                                </li>

                                                <li>

                                                    <!-- Highlight -->

                                                        <article class="box highlight">

                                                            <header>

                                                                <h3 style="color:#00C; font-size:30px"><a href="#">CONCLUSIONES Y REFERENCIAS</a></h3>

                                                            </header>

                                                            <a href="#" class="image left"><img src="images/descarga.jpg" alt="" /></a><br>

                                                            <p style="color:#000">Concluyendo este método es conveniente cuando se trata de  ordenar arreglos estáticos grandes a diferencia de otros métodos con el Quicksort y el Mergesort. Heapsort compite primariamente con Quicksort, otro método de ordenamiento muy eficiente para propósitos en general.</p>

                                                            <ul class="actions">

                                                                <li><a href="no-sidebar.html" class="button icon fa-file">VER MÁS</a></li>

                                                            </ul>

                                                        </article>

                                                </li>

                                            </ul>

                                        </section>

                                </div>

                        </div>

                    </div>

                </section>

            <!-- Footer -->

                <section id="footer">

                    <div id="copyright" class="container">

                        <ul class="links">

                            <li>&copy; Untitled. All rights reserved.</li><li>Design: <a href="http://html5up.net">HTML5 UP</a></li>

                        </ul>

                    </div>

                </section>

        </div>

        <!-- Scripts -->

            <script src="assets/js/jquery.min.js"></script>

            <script src="assets/js/jquery.dropotron.min.js"></script>

            <script src="assets/js/browser.min.js"></script>

            <script src="assets/js/breakpoints.min.js"></script>

            <script src="assets/js/util.js"></script>

            <script src="assets/js/main.js"></script>

    </body>

</html>

HeapSort

<!DOCTYPE HTML>

<!--

    Strongly Typed by HTML5 UP

    html5up.net | @ajlkn

    Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)

-->

<html>

<head>

    <title>HEAPSORT</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />

    <link rel="stylesheet" href="assets/css/main.css" />

    <!--VENTAJAS DESLIZANTES-->

    <style>

        .accordion {

            background-color:#2970ca;

            color: #444;

            cursor: pointer;

            padding: 18px;

            width: 100%;

            border: none;

            text-align: left;

            outline: none;

            font-size: 15px;

            transition: 0.4s;

        }

        .active,

        .accordion:hover {

            background-color: #ccc;

        }

        .accordion:after {

            content: '\002B';

            color: #777;

            font-weight: bold;

            float: right;

            margin-left: 5px;

        }

        .active:after {

            content: "\2212";

        }

        .panel {

            padding: 0 18px;

            background-color: white;

            max-height: 0;

            overflow: hidden;

            transition: max-height 0.2s ease-out;

        }

    </style>

</head>

<body class="left-sidebar is-preload">

    <div id="page-wrapper">

        <!-- Header -->

        <section id="header">

            <div class="container">

                <!-- Logo --><br><br><br><br>

                <h1 id="logo"><a href="index.html">ALGORITMO HEAPSORT</a></h1>

                <p>Es un algoritmo de ordenación no recursivo.</p>

                <!-- Nav -->

                <nav id="nav">

                    <ul>

                        <li><a style="color:rgb(252, 249, 249)" class="icon fa-home"

                                href="index.html"><span>INTRODUCCION</span></a></li>

                        <LI><a style="color:rgb(250, 245, 245)" class="icon fa-cog" href="left-sidebar.html"><span>HEAP

                                    SORT</span></a></li>

                                    <li><a style="color:aliceblue" class="icon fa-question-circle"

                                        href="cuestionario.html"><span>Cuestionario</span></a></li>

                        <!-- <li><a style="color:rgb(243, 234, 234)" class="icon fa-retweet"

                                href="right-sidebar.html"><span>FUNCIONAMIENTO</span></a></li>

                        <li><a style="color:rgb(238, 232, 232)" class="icon fa-sitemap"

                                href="no-sidebar.html"><span>FUENTES</span></a></li> -->

                    </ul>

                    <img src="images/wiseregistrycleaner-software-image.png" width="250" height="200">

                </nav>

            </div>

        </section>

        <!-- Main -->

        <section id="main">

            <div class="container">

                <div class="row">

                    <!-- Sidebar -->

                    <div id="sidebar" class="col-4 col-12-medium">

                        <!-- Highlights -->

                        <section>

                            <ul class="divided">

                                <li>

                                    <!-- Highlight -->

                                    <article class="box highlight">

                                        <header>

                                            <h3 style="color:#F00; font-size:20px; font-weight:bold"><a

                                                    href="#">APLICACIONES</a></h3>

                                        </header>

                                        <a href="#" class="image left"><img src="images/bombilla.gif" alt="" /></a>

                                        <p>Una de las más grandes aplicaciones de Heapsort es construir colas de

                                            prioridad con la idea que busqueda los procesos que llevan la mayor carga de

                                            prioridad dado una gran cantidad de procesos por hacer.<br>En esencia las

                                            aplicaciones de Heapsort son las que traten de sobre ordenar una lista de

                                            elementos.</p>

                                    </article>

                                </li>

                                <li>

                                    <!-- Highlight -->

                                    <article class="box highlight">

                                        <header>

                                            <h3 style="color:#939; font-size:20px; font-weight:bold"><a

                                                    href="#">VENTAJAS</a></h3>

                                        </header>

                                        <a href="#" class="image left"></a>

                                        <p>Usar este método de Ordenamiento trae consigo diversas ventajas con respecto

                                            a los otros métodos, dichas características están en la tabla a

                                            continuación:</p>

                                        <button class="accordion">VENTAJA 1</button>

                                        <div class="panel">

                                            <p style="color:#000; font-weight:bold; font-size:20px"><br>Su desempeño es

                                                en promedio tan bueno como el Quicksort y se comporta mejor que este

                                                último en los peores casos.</p>

                                        </div>

                                        <button class="accordion">VENTAJA 2</button>

                                        <div class="panel">

                                            <p style="color:#000; font-weight:bold; font-size:20px"><br> Funciona

                                                efectivamente con datos desordenados.</p>

                                        </div>

                                        <button class="accordion">VENTAJA 3</button>

                                        <div class="panel">

                                            <p style="color:#000; font-weight:bold; font-size:20px"><br> No utiliza

                                                memoria adicional.</p>

                                        </div>

                                    </article>

                                </li>

                                <li>

                                    <!-- Highlight -->

                                    <article class="box highlight">

                                        <header>

                                            <h3 style="color:#939; font-size:20px; font-weight:bold"><a

                                                    href="#">DESVENTAJAS</a></h3>

                                        </header>

                                        <a href="#" class="image left"></a>

                                        <p>Usar este método de Ordenamiento trae consigo diversas desventajas con

                                            respecto a los otros métodos, dichas características están en la tabla a

                                            continuación:</p>

                                        <button class="accordion">DESVENTAJA 1</button>

                                        <div class="panel">

                                            <p style="color:#000; font-weight:bold; font-size:20px"><br> Aunque el

                                                Heapsort tiene un mejor desempeño general que cualquier otro método

                                                presentado de clasificación interna, es bastante complejo de programar.

                                            </p>

                                        </div>

                                        <button class="accordion">DESVENTAJA 2</button>

                                        <div class="panel">

                                            <p style="color:#000; font-weight:bold; font-size:20px"><br> No es estable.

                                                Es un método complejo</p>

                                        </div>

                                    </article>

                                </li>

                            </ul>

                        </section>

                    </div>

                    <!-- Content -->

                    <div id="content" class="col-8 col-12-medium imp-medium">

                        <!-- Post -->

                        <article class="box post">

                            <header>

                                <h2 style="color:#000"><strong>DEFINICIÓN</strong> <br />

                                </h2>

                            </header>

                            <span class="image featured"><img src="images/algorithm-010.png" height="350"

                                    alt="" /></span>

                            <h3></h3>

                            <p style="color:#000; font-size:20px">

                                Este método es una variante del método por selección, donde la búsqueda del elemento

                                máximo de un arreglo se realiza mediante técnicas basadas en la construcción de un

                                montículo.<br>Puede ser un montículo ascendente o descendente.</p>

                            <p style="color:#000; font-size:20px">

                                Es un método de ordenamiento basado con comparación, usa el Montículo o Heap como

                                estructura de datos. Este método es más lento que otros métodos, pero es más eficaz en

                                escenarios más rigurosos.<br><br>

                                Se define como un metodo <strong>

                                    No recursivo, No Estable y con Complejidad Computacional.</strong>

                            </p>

                            <h3 style="color:#390; font-size:24px">PROPIEDADES</h3>

                            <p>

                            <ul>

                                <li style="color:#000; font-size:20px">

                                    Fue el primer algoritmo que surgió de los que ordenan en <strong> O(n log

                                        n)</strong>.

                                </li>

                                <li style="color:#000; font-size:20px">

                                    Ordena en el lugar. No utiliza memoria auxiliar

                                </li>

                                <li style="color:#000; font-size:20px">

                                    No es estable.

                                </li>

                                <center>

                                    <img src="images/formula.svg" width="100">

                                </center>

                            </ul>

                            </p>

                            <h3 style="color:#00C; font-size:24px">Montículo</h3>

                            <p>

                            <article class="box highlight">

                                <a href="#" class="image left"><img src="images/Binary-Heap-1b08b627.png" alt="" /></a>

                                <p style="color:#000; font-size:20px"">

                                                    <br>Es una estructura de datos del tipo árbol binario. Este árbol                                                          binario tiene que ser completo, en otras palabras, cada nivel                                                          debe de estar lleno con excepción del ultimo nivel, en el último                                                          nivel, los hijos debe recargarse hacia un mismo lado,                                                          generalmente hacia el lado izquierdo, así como se muestra en la                                                        imagen de la derecha.

                                                    </p>

                                                        </article></p>

                                        </article>

                              </div>

                        </div>

                    </div>

                </section>

            <!-- Footer -->

                <section id="footer">

                                <div class="container">

                                    <header>

                                        <br>

                                        <h2><strong>VER TAMBIEN:</strong></h2>

                                    </header>

                                    <div class="row">

                                        <div class="col-4 col-6-medium col-12-small">

                                            <!-- Feature -->

                                            <img src="images/mtodo-de-heapsort-13-638.jpg" width="400" height="240">

                                            <br>

                                            <br>

                                            <br>

                                            <header>

                                                <h3 style="color:#FFF; font-size:20px; text-align:center"><a

                                                        href="https://es.slideshare.net/DigitalXlien/mtodo-de-heapsort">CÓMO

                                                        SE CONSTRUYE?</a></h3>

                                            </header><br>

                                            <p style="color:#000"><strong> El método de Heapsort</strong> es un

                                                algoritmo de ordenamiento también llamado ordenamiento por montículos.

                                                Esta presentación tiene la explicación de como funciona el método

                                                utilizando montículos máximos.</p>

                                        </div>

                                        <div class="col-3 col-7-medium col-12-small">

                                        </div>

                                        <div class="col-4 col-7-medium col-12-small">

                                            <!-- Feature -->

                                            <section>

                                                <a href="https://www.youtube.com/watch?v=MtQL\_ll5KhQ"

                                                    class="image featured">

                                                    <video id="video1" width="400" controls>

                                                        <source src="VIDEOS/Heap Sort  GeeksforGeeks.mp4"

                                                            type="video/mp4">

                                                    </video>

                                                </a>

                                                <header>

                                                    <h3 style="color:#FFF; font-size:20px; text-align:center"><a

                                                            href="https://www.geeksforgeeks.org/heap-sort/">TECNICA DE

                                                            CLASIFICACIÓN HEAPSORT</a></h3>

                                                </header>

                                                <p style="color:#000"><strong>El algoritmo de clasificación

                                                        Heap</strong> tiene usos limitados porque Quicksort y Mergesort

                                                    son mejores en la práctica. Sin embargo, la estructura de datos Heap

                                                    en sí misma es enormemente utilizada.</p>

                                            </section>

                                        </div>

                                    </div>

                                    <div id="copyright" class="container">

                                        <ul class="links">

                                            <li>&copy; Untitled. All rights reserved.</li>

                                            <li>Design: <a href="http://html5up.net">HTML5 UP</a></li>

                                        </ul>

                                    </div>

        </section>

    </div>

    <!-- Scripts -->

    <script src="assets/js/jquery.min.js"></script>

    <script src="assets/js/jquery.dropotron.min.js"></script>

    <script src="assets/js/browser.min.js"></script>

    <script src="assets/js/breakpoints.min.js"></script>

    <script src="assets/js/util.js"></script>

    <script src="assets/js/main.js"></script>

    <!-- VENTAJAS DESLIZANTES -->

    <script>

        var acc = document.getElementsByClassName("accordion");

        var i;

        for (i = 0; i < acc.length; i++) {

            acc[i].addEventListener("click", function () {

                this.classList.toggle("active");

                var panel = this.nextElementSibling;

                if (panel.style.maxHeight) {

                    panel.style.maxHeight = null;

                } else {

                    panel.style.maxHeight = panel.scrollHeight + "px";

                }

            });

        }

    </script>

</body>

</html>

**Cuestionario**

<!DOCTYPE HTML>

<!--

    Strongly Typed by HTML5 UP

    html5up.net | @ajlkn

    Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)

-->

<html>

<head>

    <title>ALGORITMOS</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />

    <link rel="stylesheet" href="assets/css/main.css" />

</head>

<body class="homepage is-preload">

    <div id="page-wrapper">

        <!-- Header -->

        <section id="header">

            <div class="container">

                <!-- Logo --><br><br><br><br>

                </nav>

                </nav>

                <h1 id="logo"><a href="index.html">Cuestionario<br><br></a></h1>

                <!-- Nav -->

                <nav id="nav">

                    <ul>

                        <li><a style="color:aliceblue" class="icon fa-question-circle"

                            href="cuestionario.html"><span>Cuestionario</span></a></li>

                        <li><a style="color:aliceblue" class="icon fa-home"

                                href="index.html"><span>INTRODUCCION</span></a></li>

                        <LI><a style="color:aliceblue" class="icon fa-cog" href="left-sidebar.html"><span>HEAP

                                    SORT</span></a></li>

                    </ul>

                    <img src="images/logo.png" width="700" height="200">

            </div>

        </section>

        <!-- Features -->

        <section id="features">

            <div class="container">

                <header>

                    <h2 style="color:#000">Preguntas Importantes</h2>

                </header>

                <div class="row aln-center">

                </div>

            </div>

        </section>

        <!-- Banner -->

        <section id="banner">

            <div class="container">

                <p><strong>Querido Visitante</strong></p>

                <p>Los métodos de ordenamiento son ampliamente usados en el desarrollo de software,

                    debido a que posibilitan tomar un grupo de datos y ordenarlos

                    de manera secuencial en un tipo de datos de agrupación o arreglo.</p><br>

                <p><strong>Esta serie de preguntas te ayudarán a saber cuanto has aprendido</strong></p>

            </div>

        </section>

        <!-- Main -->

        <section id="main">

            <div class="container">

                <div class="row">

                    <!-- Content -->

                    <div id="content" class="col-8 col-12-medium">

                        <article class="box post">

                            <header>

                                <h2 style="font-size:40px; color:#90C"><a><strong>Cuestionario</strong></a></h2>

                            </header>

                            <a href="#" class="image featured"><img src="images/survey-2316468\_1280.png" alt="" /></a>

                            <h3 style="color:#00C">Primera Parte</h3>

                            <p style="color:#000">

                                <strong>1.¿ Que son los métodos de ordenamiento?<br><br>

                                    <p>Un algoritmo de ordenamiento es un algoritmo que pone elementos de una

                                         lista o un vector en una secuencia dada por una relación de orden,

                                          es decir, el resultado de salida ha de ser una permutación o reordenamiento de la entrada que satisfaga la relación de orden dada. </p>

                                <strong>2. </strong> ¿Que es el metodo de ordenamiento Heapsort?<br><br>

                                <p>

                                    Basa su funcionamiento en una propiedad de los montículos, por la cual, la cima contiene siempre el menor elemento (o el mayor, según se haya definido el montículo) de todos los almacenados en él. El algoritmo, después de cada extracción, recoloca en el nodo raíz o cima, la última hoja por la derecha del último nivel.

                                </p>

                                <strong>3. </strong> ¿Cuales métodos de ordenamiento existen?<br><br>

                                Bubblesort,Cocktail sort,Insertion sort son los mas comunes

                                <strong>4. </strong> ¿Cual es el método de ordenamiento mas eficiente?<br><br>

                                <p>Quicksort</p>

                                <strong>5. </strong> ¿Qué condiciones debecumplir el árbol de ordenamiento?<br><br>

                                <p ><strong > Árbol binario que cumple ciertas condiciones:</strong><br><br>

                                    <strong>1. </strong> El valor de cada nodo es mayor o igual a los valores almacenados en cada uno de sus hijos.<br><br>

                                    <strong>2. </strong> El árbol esta perfectamente balanceado y las hojas del ultimo nivel están todas en las<br>

                                        posiciones en el extremo izquierdo.</p>

                                <strong>6. </strong> ¿Cual es una aplicación del algoritmo heapsort?<br><br>

                                        <p>Una de las más grandes aplicaciones de Heapsort es construir colas de prioridad con la idea que busqueda los procesos que llevan la mayor carga de prioridad dado una gran cantidad de procesos por hacer.</p>

                                <strong>7. </strong> Una Ventaja del algoritmo heapsort<br><br>

                                        <p>Su desempeño es en promedio tan bueno como el Quicksort y se comporta mejor que este último en los peores casos.</p>

                            </p>

                                                       <ul class="actions">

                                <li><a href="left-sidebar.html" class="button icon fa-file">CONTINUAR LEYENDO</a></li>

                            </ul>

                        </article>

                    </div>

                    <!-- Sidebar -->

                    <div id="sidebar" class="col-4 col-12-medium">

                        <!-- Excerpts -->

                        <section>

                            <ul class="divided">

                            </ul>

                        </section>

                    </div>

                </div>

            </div>

        </section>

        <!-- Footer -->

        <section id="footer">

            <div id="copyright" class="container">

                <ul class="links">

                    <li>&copy; Untitled. All rights reserved.</li>

                    <li>Design: <a href="http://html5up.net">HTML5 UP</a></li>

                </ul>

            </div>

        </section>

    </div>

    <!-- Scripts -->

    <script src="assets/js/jquery.min.js"></script>

    <script src="assets/js/jquery.dropotron.min.js"></script>

    <script src="assets/js/browser.min.js"></script>

    <script src="assets/js/breakpoints.min.js"></script>

    <script src="assets/js/util.js"></script>

    <script src="assets/js/main.js"></script>

</body>

</html>

Se crearon los estilos css en el archivo llamado main.css

@import url("font-awesome.min.css");

@import url("https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,600|Arvo:700");

/\*

    Strongly Typed by HTML5 UP

    html5up.net | @ajlkn

    Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)

\*/

html, body, div, span, applet, object,

iframe, h1, h2, h3, h4, h5, h6, p, blockquote,

pre, a, abbr, acronym, address, big, cite,

code, del, dfn, em, img, ins, kbd, q, s, samp,

small, strike, strong, sub, sup, tt, var, b,

u, i, center, dl, dt, dd, ol, ul, li, fieldset,

form, label, legend, table, caption, tbody,

tfoot, thead, tr, th, td, article, aside,

canvas, details, embed, figure, figcaption,

footer, header, hgroup, menu, nav, output, ruby,

section, summary, time, mark, audio, video {

    margin: 0;

    padding: 0;

    border: 0;

    font-size: 100%;

    font: inherit;

    vertical-align: baseline;}

article, aside, details, figcaption, figure,

footer, header, hgroup, menu, nav, section {

    display: block;}

body {

    line-height: 1;

}

ol, ul {

    list-style: none;

}

blockquote, q {

    quotes: none;

}

    blockquote:before, blockquote:after, q:before, q:after {

        content: '';

        content: none;

    }

table {

    border-collapse: collapse;

    border-spacing: 0;

}

body {

    -webkit-text-size-adjust: none;

}

mark {

    background-color: transparent;

    color: inherit;

}

input::-moz-focus-inner {

    border: 0;

    padding: 0;

}

input, select, textarea {

    -moz-appearance: none;

    -webkit-appearance: none;

    -ms-appearance: none;

    appearance: none;

}

/\* Basic \*/

    html {

        box-sizing: border-box;

    }

    \*, \*:before, \*:after {

        box-sizing: inherit;

    }

    body {

        background: #e8f1f2;

    }

        body.is-preload \*, body.is-preload \*:before, body.is-preload \*:after {

            -moz-animation: none !important;

            -webkit-animation: none !important;

            -ms-animation: none !important;

            animation: none !important;

            -moz-transition: none !important;

            -webkit-transition: none !important;

            -ms-transition: none !important;

            transition: none !important;

        }

    body, input, textarea, select {

        font-family: 'Source Sans Pro';

        font-weight: 300;

        color: #777;

        line-height: 1.65em;

        font-size: 15pt;

    }

    h1, h2, h3, h4, h5, h6 {

        font-weight: 600;

        text-transform: uppercase;

        color: #2f2d2e;

    }

    h1 a, h2 a, h3 a, h4 a, h5 a, h6 a {

        color: inherit;

        text-decoration: none;

        border: 0;

    }

    h2 {

        font-size: 1.65em;

        font-weight: 400;

        letter-spacing: 4px;

        margin: 0 0 1.5em 0;

        line-height: 1.75em;

    }

    h3 {

        font-size: 1em;

        letter-spacing: 2px;

        margin: 0 0 1.25em 0;

    }

    a {

        -moz-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

        -webkit-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

        -ms-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

        transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

        color: #666;

        text-decoration: none;

        border-bottom: solid 1px #ddd;

    }

        a strong {

            -moz-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

            -webkit-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

            -ms-transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

            transition: color 0.25s ease-in-out, border-bottom-color 0.25s ease-in-out;

        }

        a:hover {

            color: #13293d;

            border-bottom-color: rgba(255, 255, 255, 0);

        }

            a:hover strong {

                color: #13293d;

            }

    strong, b {

        font-weight: 600;

        color: #666;

    }

    em, i {

        font-style: italic;

    }

    sub {

        position: relative;

        top: 0.5em;

        font-size: 0.8em;

    }

    sup {

        position: relative;

        top: -0.5em;

        font-size: 0.8em;

    }

    hr {

        border: 0;

        border-top: solid 1px #ddd;

    }

    blockquote {

        border-left: solid 0.5em #ddd;

        padding: 1em 0 1em 2em;

        font-style: italic;

    }

    p {

        text-align: justify;

        margin-bottom: 2em;

    }

    ul, ol, dl, table, blockquote {

        margin-bottom: 2em;

    }

    br.clear {

        clear: both;

    }

**Resultados y Discusión**

Vista final de la página de inicio



