Análisis de eficiencia

|  |  |  |
| --- | --- | --- |
| page3image7869936 | **MaxColaCP** | **MaxHeapCP** |
| ¿Cuándo ocurre el peor caso de agregar()? | No hay peor caso con la implementación propuesta. Se le agrega a la última posición del arreglo. | Ya que se debe hacer un swim, y la altura del heap es log N, el peor caso depende de la prioridad del elemento a agregar. Si este elemento tiene la prioridad más alta, debe recorrer todo el heap y intercambiar con logN elementos. |
| page3image14232944Complejidad peor caso de agregar() | page3image7940656page3image7889280O(1) | page3image12068608page3image16293072page3image81330992page3image81327872O(logN) |
| ¿Cuándo ocurre el mejor caso de agregar() | Misma razón. | El mejor caso sería agregar en un heap de tamaño 0. (El swim no tiene que realizar ninguna acción) O agregar un elemento con la prioridad más baja. |
| page3image16289744Complejidad mejor caso de agregar() | page3image7882000page3image7894064O(1) | page3image81264848page3image7871600page3image5887184page3image12188416O(1) |
| Tiempo promedio de agregar() (Promedio de agregar 200,000 datos a una cola vacía) | 22 milisegundos | 33 milisegundos |
| page3image12163040¿Cuándo ocurre el peor caso de sacarMax ()? | page3image12164704El peor caso ocurre cuando el elemento máximo del arreglo es el primero posisionado (o el primero en haber ingresado). | NIDEA |
| page3image7953344Complejidad peor caso de sacarMax () | page3image12123728page3image12185296O(n) | page3image14237520page3image18355488page3image12178640page3image14188016 |
| ¿Cuándo ocurre el mejor caso de sacarMax ()? | El elemento maximo es el último posisionado o el último en entrar. | NIDEA |
| page3image18364848Complejidad mejor caso de sacarMax () | page3image18378368page3image18378160O(1) | page3image12117488page3image12119152page3image18365472page3image5799824 |
| Tiempo promedio de sacarMax () (Promedio de sacar 200,000 datos de una cola con 200,000 datos) | 59661 milisegundos | 118 milisegundos |