|  |  |  |
| --- | --- | --- |
| TAD Red-Black Tree | | |
| Image result for redblack tree  L= | | |
| INVARIANTES | | |
| • La raíz del árbol debe ser de color negro  • nodo izquierdo datos menores  • nodo derecho datos mayores  • Los nodos de color rojo solo pueden tener hijos nodos de color negro  • Los hijos de cada nodo hoja deben ser null.  • Cada nodo nuevo que se agrega debe de ser color rojo  • Los nodos deben ser de color rojo o negro  • La altura negra de un nodo debe ser igual tanto por izquierda como por derecha. | | |
| OPERACION | ENTRADA | SALIDA |
| Crear |  | L |
| Buscar | A | nodo |
| Insertar | nodo | L |
| Eliminar | A | L |

*\*A= tipo de dato variable.*

|  |
| --- |
| Buscar(variable) |
| Dado una variable buscar si algún elemento tiene exactamente el mismo valor, si es así lo retorna. |
| {Pre: variable!=null&&tree.getRoot()!=null} |
| {Post: si el elemento está en el árbol lo devuelve si no está lanza una excepción} |

|  |
| --- |
| Crear() |
| Genera un árbol rojo y negro vacío. |
| {Pre: tree.getRoot()==null} |
| {Post: true}  Image result for black point png |

|  |
| --- |
| Insertar(nodo) |
| Inserta un nodo dado al árbol rojo y negro. |
| {Pre: nodo!=null&&tree.getRoot()!=null} |
| {Post: se agrega el nodo}  rbtrecoloring.png  *\*example of random case* |

|  |
| --- |
| Eliminar(variable) |
| Dado una variable buscar si algún elemento tiene exactamente el mismo valor, si es así lo elimina. |
| {Pre: variable!=null&&tree.getRoot()!=null} |
| {Post: si el elemento está en el árbol lo elimina si no está lanza una excepción}    *\*example of random case* |

|  |  |  |
| --- | --- | --- |
| TAD AVL | | |
| Image result for avl tree  L= | | |
| INVARIANTES | | |
| • factor de balanceo=[-1,0,1]  • nodo izquierdo datos menores  • nodo derecho datos mayores  • debe estar balanceado | | |
| OPERACION | ENTRADA | SALIDA |
| Crear |  | L |
| Buscar | A | nodo |
| Insertar | nodo | L |
| Eliminar | A | L |

*\*A= tipo de dato variable.*

|  |
| --- |
| Crear() |
| Genera un árbol AVL vacío. |
| {Pre: tree.getRoot()==null} |
| {Post: true}  Image result for black point png |

|  |
| --- |
| Insertar(nodo) |
| Inserta un nodo dado al árbol AVL. |
| {Pre: node!=null&&tree.getRoot()!=null} |
| {Post: se agrega el nodo}    *\*example of random case* |

|  |
| --- |
| Buscar(variable) |
| Dado una variable buscar si algún elemento tiene exactamente el mismo valor, si es así lo retorna. |
| {Pre: variable!=null&&tree.getRoot()!=null} |
| {Post: si el elemento está en el árbol lo devuelve si no está lanza una excepción} |

|  |
| --- |
| Eliminar(variable) |
| Dado una variable buscar si algún elemento tiene exactamente el mismo valor, si es así lo elimina. |
| {Pre: variable!=null&&tree.getRoot()!=null} |
| {Post: si el elemento está en el árbol lo elimina si no está lanza una excepción}    *\*example of random case* |