

1.

TAD Árbol Binario de búsqueda (ABB)		
ABB = {Raiz = $\langle r \rangle$ }		
{inv: $r \in \text{Nodo}$ }		
Operaciones Primitivas		
- CrearArbol	Nodo	ABB
- AgregarNodo	ABB x Nodo	ABB
- EliminarNodo	ABB x Nodo	ABB
- BuscarNodo	ABB x Objeto	Nodo
- BuscarNodo	ABB X Objeto x Nodo	Nodo
- Minimo	ABB x Nodo	Nodo
- ObtenerRaiz	ABB	Nodo

CrearArbol(r)

“Crea un árbol binario de búsqueda asignando como raíz el nodo r ”

{pre: $r \in \text{Nodo}$ }

{post: $ABB = \text{Raiz} = \langle r \rangle$ }

AgregarNodo(abb, n)

“Agrega un nodo n al árbol binario de búsqueda”

{pre: $n \in \text{Nodo}$ }

{post: $ABB = \text{Raiz} = \langle r \rangle$ }

EliminarNodo(abb, n)

“Elimina un nodo n del árbol binario de búsqueda”

{pre: $n \in \text{Nodo}$ }

{post: $ABB = \text{Raiz} = \langle r \rangle$ }

BuscarNodo(abb, o)

“Retorna un nodo buscado n que tenga un objeto o del árbol binario de búsqueda”

{pre: $o \in \text{Objeto}$ }

{post: $\text{Nodo} = \text{Objeto} = \langle o \rangle$, Derecho = $\langle d \rangle$, Izquierdo = $\langle i \rangle$, Padre = $\langle p \rangle$ }

>}

BuscarNodo(abb, o, n)

“Retorna por medio de recursividad un nodo buscado n del árbol binario de búsqueda que tenga el objeto o ”

{pre: $n \in \text{Nodo} \wedge o \in \text{Objeto}$ }

{post: $\text{Nodo} = \text{Objeto} = \langle o \rangle$, Derecho = $\langle d \rangle$, Izquierdo = $\langle i \rangle$, Padre = $\langle p \rangle$ }

Minimo(abb, n)

“Busca el nodo con el mínimo objeto en una rama de nodos n ”

{pre: $n \in \text{Nodo}$ }

{post: $\text{Nodo} = \text{Objeto} = \langle o \rangle$, Derecho = $\langle d \rangle$, Izquierdo = $\langle i \rangle$, Padre = $\langle p \rangle$ }

ObtenerRaiz(abb)

“Retorna la raíz del árbol binario de búsqueda”

{pre: $r \in \text{Nodo} \wedge abb \neq \text{nulo}$ }

{post: $\langle r \rangle$ }

TAD Nodo

$\text{Nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle d \rangle, \text{Izquierdo} = \langle i \rangle, \text{Padre} = \langle p \rangle\}$

{inv: $d, i, p \in \text{Nodo} \wedge \text{objeto} > i.\text{objeto} \wedge \text{objeto} < d.\text{objeto} \wedge o \neq \text{nulo}$ }

Operaciones Primitivas

- CrearNodo	Objeto	Nodo
- CambiarValor	Nodo x Objeto	Nodo
- MostrarValor	Nodo	Objeto
- DarIzquierdo	Nodo	i
- DarDerecho	Nodo	d
- DarPadre	Nodo	p
- AgregarDerecho	Nodo x Nodo	Nodo
- AgregarIzquierdo	Nodo x Nodo	Nodo
- AgregarPadre	Nodo x Nodo	Nodo

CrearNodo(o)

“Crea un nodo con un objeto inicial”

{pre: $o \neq \text{nulo}$ }

{post: $\text{nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle \text{nulo} \rangle, \text{Izquierdo} \langle \text{nulo} \rangle, \text{Padre} = \langle \text{nulo} \rangle\}$ }

CambiarValor(o)

“Cambia el objeto que posee el nodo”

{pre: $o \neq \text{nulo} \wedge \text{nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle d \rangle, \text{Izquierdo} \langle i \rangle, \text{Padre} = \langle p \rangle\}$ }

{post: $\text{nodo} = \{\text{Objeto} = o, \text{Derecho} = \langle d \rangle, \text{Izquierdo} \langle i \rangle, \text{Padre} = \langle p \rangle\}$ }

MostrarValor()

“Muestra el objeto que posee el nodo”

{pre: $\text{nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle d \rangle, \text{Izquierdo} \langle i \rangle, \text{Padre} = \langle p \rangle\}$ }

{post: $\langle o \rangle$ }

DarIzquierdo()

“Retorna el nodo izquierdo del nodo actual”

{pre: $\text{nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle d \rangle, \text{Izquierdo} \langle i \rangle, \text{Padre} = \langle p \rangle\}$ }

{post: $\langle i \rangle$ }

DarDerecho()

“Retorna el nodo derecho del nodo actual”

{pre: $\text{nodo} = \{\text{Objeto} = \langle o \rangle, \text{Derecho} = \langle d \rangle, \text{Izquierdo} \langle i \rangle, \text{Padre} = \langle p \rangle\}$ }

{post: $\langle d \rangle$ }

DarPadre()

“Retorna el nodo padre del nodo actual”

{pre: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle d \rangle, Izquierdo \langle i \rangle, Padre = \langle p \rangle\}$

{post: $\langle p \rangle$ }

AgregarDerecho(nodo, der)

“Añade al *Derecho* de *nodo* el nodo *der* ingresado por parámetro ”

{pre: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle nulo \rangle, Izquierdo \langle i \rangle$
 $\wedge der \in Nodo \}$

{post: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle der \rangle, Izquierdo \langle i \rangle, Padre = \langle p \rangle\}$

AgregarIzquierdo(nodo, izq)

“Añade al *Izquierdo* de *nodo* el nodo *izq* ingresado por parámetro ”

{pre: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle d \rangle, Izquierdo \langle nulo \rangle$
 $\wedge izq \in Nodo \}$

{post: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle d \rangle, Izquierdo \langle izq \rangle, Padre = \langle p \rangle\}$

AgregarPadre(nodo, pad)

“Añade al *Padre* de *nodo* el nodo *pad* ingresado por parámetro ”

{pre: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle d \rangle, Izquierdo \langle nulo \rangle$
 $\wedge pad \in Nodo \}$

{post: $nodo = \{Objeto = \langle o \rangle, Derecho = \langle d \rangle, Izquierdo \langle izq \rangle, Padre = \langle pad \rangle\}$

2. [Repositorio de GitHub](#)