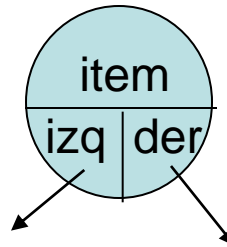


# Arbol Binario de Búsqueda

```
class nodo
{
    int item;
    nodo *izq;
    nodo *der;
public:
    int obteneritem(void)
    {
        return(item);
    }
    void cargaritem(int xitem)
    {
        item=xitem;
    }
    nodo* cargarizq(nodo* punt)
    {
        izq=punt;
        return(izq);
    }
}
```



```
nodo* obtenerizq(void)
{
    return(izq);
}
nodo* cargarder(nodo* punt)
{
    der=punt;
    return(der);
}
nodo* obtenerder(void)
{
    return(der);
}
};
```

# Arbol Binario de Búsqueda

```
class arbol
{
    nodo *raiz;
public:
//CONSTRUCTOR
    arbol(nodo* xpr=NULL): raiz(xpr)
    {}
// VACIO
    int Vacio(void)
    {
        return (raiz==NULL);
    }
/// RAIZ
    nodo* obteneraiz(void)
    {
        return(raiz);
    }
//MOSTRAR RAIZ
    int mostrarraiz(void)
    {
        return(raiz->obteneritem());
    }
}
```

```
//INSERTAR RAIZ
void insertarraiz(int x)
{
    nodo *aux;
    aux=new(nodo);
    aux->cargaritem(x);
    aux->cargarizq(NULL);
    aux->cargarder(NULL);
    raiz=aux;
}
```

```

void main(void)
{   arbol a;
    int elem, op;
    do
    {   clrscr();
        cout<<"1-Insertar un elemento en el arbol\n";
        cout<<"2-Mostar el arbol\n";
        cout<<"3-Mostrar Raiz\n";
        cout<<"5-Suprimir\n";
        cout<<"6-Buscar\n";
        cout<<"7-Altura\n";
        cout<<"8-Nivel de un nodo\n";
        cout<<"9-Contar Nodos\n";
        cout<<"0-Salir\n";
        cout<<"Opcion:";    cin>>op;

switch (op)
{   case 1: {   cout<<"Ingrese un elemento\n";
                cin>>elem;
                if (a.Vacio())
                    {a.insertarraiz(elem);}
                else
                    {a.insertar(elem,a.obtenerraiz());}
                break; }

    case 2:
        {   a.mostrar(a.obtenerraiz());
            getchar(); break; }

    case 3:
        {   cout<<"RAIZ "<<a.mostrarraiz();    .....}

    case 5:
        {
            a.Borrar(12);
            getchar();    break; }
}
}

```

```

case 6:
{
    int ele;
    cout << "Ingrese elemento a buscar";
    cin >> ele;
    if (a.Buscar(ele))
        cout << "lo encontré";
    else
        cout << "no lo encontré";
    getchar();
    break;
}

case 7:
{   nodo* raiz=a.obtenerraiz();
    m1=0;m2=0;
    cout << "Altura de arbol " << a.auxAltura(raiz) << endl;
    getchar();
    break;
}

case 8:
{
    int ele;
    cout << "Ingrese elemento ";
    cin >> ele;
    cout << "Nivel del nodo " << a.Altura(ele) << endl;
    getchar();
    break;
}

case 9:
{
    cout << "N nodos: " ;
    nodo* raiz =a.obtenerraiz();
    contador=0;
    cout<< a.auxContador(raiz); // FUncción auxiliar
    getchar();
    break;
}
} // case
} //while
while (op!=0);

```