



ESCUELA POLITÉCNICA NACIONAL
ESCUELA DE FORMACIÓN DE TECNÓLOGOS



ALGORITMOS Y ESTRUCTURAS DE DATOS

ASIGNATURA:	Algoritmos y Estructuras de Datos
PROFESOR:	Ing. Loarte Byron
PERÍODO ACADÉMICO:	2019-A

DEBER

TÍTULO:

MENU DE RECORRIDO DE UN ÁRBOL

INTEGRANTE

- Brayan David Pisuña Paillacho

FECHA DE REALIZACIÓN: 8/06/2019

DE ENTREGA: 11/06/2019

CALIFICACIÓN OBTENIDA: FIRMA
DEL PROFESOR:

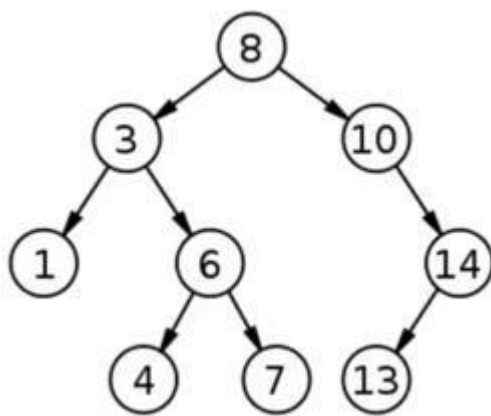
1. Recorrido de un árbol binario

Lo que se va a realizar es la creación del menú y el ingreso correspondiente de los números para que mi emprima el árbol

```
Bienvenido a los arboles binarios

Ingrese las opcion que desea realizar
1  INSERTAR NUEVO NODO
2  MOSTRAR ARBOL
3  RECORRER EL ARBOL EN PRE ORDEN
4  RECORRER EL ARBOL EN IN ORDEN
5  RECORRER EL ARBOL EN POS ORDEN
6  Salir
```

Ilustración 1 MENU DE ARBOLES BINARIOS



8 – 3 – 1 – 6 – 4 – 7 – 10 – 14 – 13

```

28 //FUNCION DE MENU
29 void menu(){
30
31 int dato , opcion, separador = 0;
32 do{
33
34 cout<<"\t\tBienvenido a los arboles binarios\n\n";
35 //con '\t\t\t' llevo un poco al centro el mensaje, y con '\n\n' salto 2 lineas hacia abajo.
36 cout<<"Ingrese las opcion que desea realizar\n";
37 cout<<"1 INSERTAR NUEVO NODO\n";
38 cout<<"2 MOSTRAR ARBOL\n";
39 cout<<"3 RECORRER EL ARBOL EN PRE ORDEN\n";
40 cout<<"4 RECORRER EL ARBOL EN IN ORDEN\n";
41 cout<<"5 RECORRER EL ARBOL EN POS ORDEN\n";
42 cout<<"6 Salir\n\n";
43 cin>>opcion;
44
45 switch(opcion)
46 {
47 case 1:
48 cout<<" digite un numero "<<endl;
49 cin>>dato;
50 insertarNodos(arbolBinario, dato);
51 system("pause");
52 break;
53 case 2:
54 cout<<"MOSTRAR ARBOL COMPLETO "<<endl;
55 mostrarArbolBinario(arbolBinario, separador);
56 mostrarArbolBinario(arbolBinario, separador);
57 system("pause");
58 break;
59 case 3:
60 cout<<"EL ARBOL EN PRE ORDEN"<<endl;
61 preorden(arbolBinario);
62 cout<<"\n";
63 system("pause");
64
65 break;
66
67 case 4:
68 cout<<"EL ARBOL EN IN ORDEN "<<endl;
69 intorden(arbolBinario);
70 cout<<"\n";
71 system("pause");
72 break;
73
74 case 5:
75 cout<<"URECORRER EL ARBOL EN POS ORDEN"<<endl;
76 postorden(arbolBinario);
77 cout<<"\n";
78 system("pause");
79 break;
80
81 case 6:
82 cout<<"Ya esta fuera"<<endl;

```

Ilustración 2 CREACION DEL MENU

```

4 struct Nodo
5 {
6     int dato;
7     Nodo *derecho;
8     Nodo *izquierdo;
9 };
10
11 Nodo *crearNodo(int n);
12 Nodo *arbolBinario = NULL;
13 void insertarNodos(Nodo *&arbolBinario, int n);
14 void mostrarArbolBinario(Nodo *&arbolBinario, int separador);
15
16 void menu();
17 void preorden (Nodo *&arbolBinario);
18 void intorden (Nodo *&arbolBinario);
19 void postorden (Nodo *&arbolBinario);

```

Ilustración 3 ESTRUCTURA Y FUNCIONES CREADAS

```

92 Nodo *crearNodo(int n){
93     Nodo *nuevo= new Nodo();
94     nuevo -> dato=n;
95     nuevo -> derecho=NULL;
96     nuevo -> izquierdo=NULL;
97     return nuevo;
98 }

```

Ilustración 4 CREAR NUEVOS NODOS

```

100 void insertarNodos(Nodo *&arbolBinario, int n){
101     if (arbolBinario == NULL)
102     {
103         Nodo *nuevo = crearNodo(n);
104         arbolBinario = nuevo;
105     }
106     else
107     {
108         int valorRaiz = arbolBinario->dato;
109         if (n<valorRaiz)
110         {
111             insertarNodos(arbolBinario->izquierdo,n);
112         }
113         else
114         {
115             insertarNodos(arbolBinario->derecho,n);
116         }
117     }
118 }

```

Ilustración 5 FUNCION PARA INSERTAR NUEVOS NODOS

```

120 void mostrarArbolBinario(Nodo *&arbolBinario, int separador){
121     if (arbolBinario==NULL)
122     {
123         return;
124     }
125     else
126     {
127         mostrarArbolBinario(arbolBinario->derecho, separador + 1);
128         for (int i=0; i<separador; i++)
129         {
130             cout<<" ";
131         }
132         cout<<arbolBinario->dato<<endl;
133         mostrarArbolBinario(arbolBinario-> izquierdo,separador+1);
134     }
135 }

```

Ilustración 6 FUNCION PARA MOSTRAR EL ARBOL BINARIO

```

137 void preorden (Nodo *&arbolBinario){
138     if (arbolBinario==NULL)
139     {
140         return;
141     }
142     else
143     {
144         cout<<arbolBinario->dato<<"-";
145         preorden(arbolBinario->izquierdo);
146         preorden(arbolBinario->derecho);
147     }
148 }

```

Ilustración 7 FUNCION PARA EL PRE ORDEN

```

150 void intorden (Nodo *&arbolBinario){
151     if (arbolBinario==NULL)
152     {
153         return;
154     }
155     else
156     {
157         intorden(arbolBinario -> izquierdo);
158         cout<<arbolBinario->dato<<" - ";
159     }
160 }
161 }

```

Ilustración 8 FUNCION PARA EL IN ORDEN

```

163 void postorden (Nodo *&arbolBinario){
164     if (arbolBinario==NULL)
165     {
166         return;
167     }
168     else
169     {
170         postorden(arbolBinario -> izquierdo);
171         postorden(arbolBinario -> derecho);
172         cout<<arbolBinario->dato<<" - ";
173     }
174 }
175 }

```

Ilustración 9 FUNCION PARA EL POST ORDEN

1. VISTA DEL MENU COMPLETAMENTE TERMINADO

```

                                Bienvenido a los arboles binarios

Ingresa la opción que desea realizar
1  INSERTAR NUEVO NODO
2  MOSTRAR ARBOL
3  RECORRER EL ARBOL EN PRE ORDEN
4  RECORRER EL ARBOL EN IN ORDEN
5  RECORRER EL ARBOL EN POS ORDEN
6  Salir

```

Ilustración 10 VISTA DE MENU

```

MOSTRAR ARBOL COMPLETO
      14
     10 13
    8   7
   6   4
  3   1
Presione una tecla para continuar . . .

```

Ilustración 11 COMO SE VE NUESTRO GRAFICO

```

EL ARBOL EN PRE ORDEN
8-3-1-6-4-7-10-14-13-
Presione una tecla para continuar . . .

```

Ilustración 12 ARBOL EN PRE ORDEN

```

EL ARBOL EN IN ORDEN
1 - 3 - 8 -
Presione una tecla para continuar . . .

```

Ilustración 13 ARBOL EN IN ORDEN

```

RECORRER EL ARBOL EN POS ORDEN
1 - 4 - 7 - 6 - 3 - 13 - 14 - 10 - 8 -
Presione una tecla para continuar . . .

```

Ilustración 14 ARBOL EN POST ORDEN

CODIGO DEL MENU : <https://github.com/BrayanPisuna/Nodos>