



Tecnológico de Monterrey

Campus Santa Fe

Programación de estructuras de datos y algoritmos fundamentales

Integrantes del equipo:

Andres Tarazona Solloa - A010233332

Profesor:

Dr. Esteban Castillo Juarez

Fecha:

3 de octubre de 2022

Número de nodos en un árbol binario:

Para conseguir el número de nodos en un árbol, solo se debe hacer un recorrido y regresar un 1 por cada nodo por el que se pasa:

```
int nodeCount(struct nodo *inicio) {
    if (inicio == NULL) {
        return 0;
    } else {
        return 1 + nodeCount(inicio->izq) + nodeCount(inicio->der);
    }
}
```

Entonces nos da lo siguiente:

Nodos: 15

Para contar los nodos pares e impares y sumar los valores, se hace un recorrido similar al del conteo de nodos, solamente cambia el if, y se regresa entonces el valor del nodo:

```
int sumaPares(struct nodo *inicio) {
    if (inicio == NULL) {
        return 0;
    } else {
        if (inicio->valor % 2 == 0)
            return inicio->valor + sumaPares(inicio->izq) + sumaPares(inicio->der);
        else
            return sumaPares(inicio->izq) + sumaPares(inicio->der);
    }
}
```

```
int sumaImpares(struct nodo *inicio) {
    if (inicio == NULL) {
        return 0;
    } else {
        if (inicio->valor % 2 != 0)
            return inicio->valor + sumaImpares(inicio->izq) + sumaImpares(inicio->der);
        else
            return sumaImpares(inicio->izq) + sumaImpares(inicio->der);
    }
}
```

Da lo siguiente:

Nodos pares: 80

Nodos impares: 84

Para imprimir los nodos hoja del árbol, solamente tenemos que hacer un recorrido y checar si cada nodo tiene hijos, si sí los tiene entonces no es una hoja, de otra forma se imprime:

```
void nodoHoja(struct nodo *inicio) {
    if (inicio == NULL) {
        return;
    } else {
        if (inicio->izq == NULL && inicio->der == NULL) {
            printf("%d ", inicio->valor);
        }
        nodoHoja(inicio->izq);
        nodoHoja(inicio->der);
    }
}
```

Imprime lo siguiente entonces:

```
Nodos hoja:
4 7 12 22
```

Finalmente para el recorrido en anchura, se calcula la altura del árbol en total, y después se crea una función para imprimir cada nivel. Después, se crea un for loop sobre el cual se itera a lo largo de la altura del árbol, y se imprime entonces cada nivel.

```
int altura(struct nodo *inicio) {
    if (inicio == NULL) {
        return 0;
    } else {
        int izq = altura(inicio->izq);
        int der = altura(inicio->der);
        if (izq > der) {
            return izq + 1;
        } else {
            return der + 1;
        }
    }
}

void imprimirNivel(struct nodo *inicio, int nivel) {
    if (inicio == NULL) {
        return;
    } else {
        if (nivel == 1) {
            printf("%d ", inicio->valor);
        } else {
            imprimirNivel(inicio->izq, nivel - 1);
            imprimirNivel(inicio->der, nivel - 1);
        }
    }
}

void imprimirNiveles(struct nodo *inicio) {
    int h = altura(inicio);
    for (int i = 1; i <= h; i++) {
        imprimirNivel(inicio, i);
    }
}
```

Se imprime entonces:

```
Operacion anchura:  
10 9 15 8 12 18 5 19 3 5 21 4 6 22 7
```