



Algoritmos y Estructuras de Datos

Cursada 2017
Redictado

Prof. Alejandra Schiavoni (ales@info.unlp.edu.ar)

Prof. Catalina Mostaccio (catty@lifa.info.unlp.edu.ar)

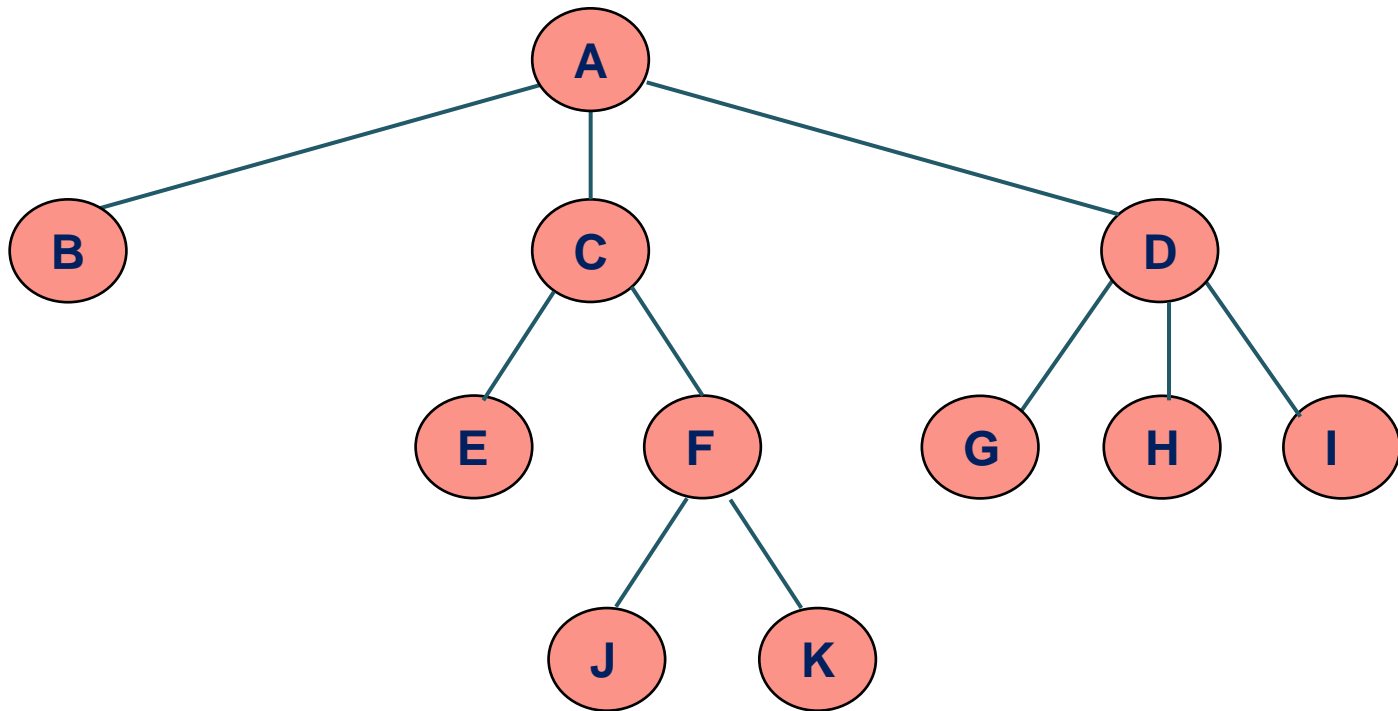
Prof. Pablo Iuliano (piuliano@info.unlp.edu.ar)

Árboles Generales

Ejercicios

- 1) ¿Cuántos **niveles** tiene el árbol?
- 2) ¿Cuántos **nodos** hay en **cada** nivel del árbol?
- 3) ¿Cuántos **nodos** hay en el nivel k del árbol?

Resolución de Ejercicios



Resolución del Ejercicio 1

Seudocódigo Ejerc1-Niveles {

q: cola de vértices;

encolar raíz **R** en q;

mientras (cola no se vacíe) {

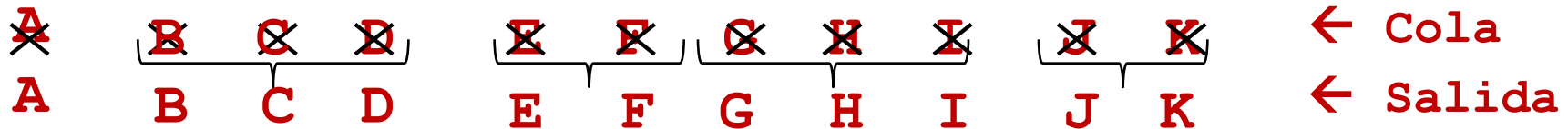
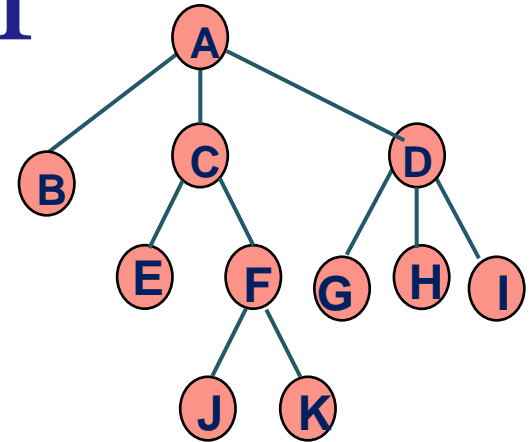
 desencolar **v** de q;

 imprimir (dato de v);

 para cada hijo **w** de **v**

 encolar **w** en q; }

}



Recorrido por niveles

Seudocódigo Recorrido_Niveles {

q: cola de vértices;

encolar raíz **R** en q;

mientras (cola no se vacíe) {

desencolar **v** de q;

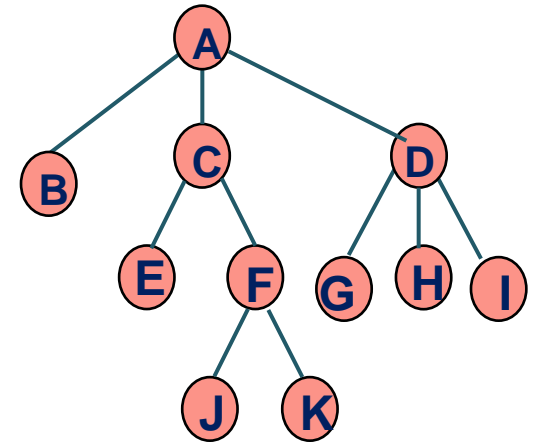
imprimir (dato de **v**);

para cada hijo **w** de **v**

encolar **w** en q;

}

}



A B C D E F G H I J K

Ejercicio

¿Cuántos **niveles** tiene el árbol?

- En el recorrido por niveles no se distingue a qué nivel pertenece cada nodo
- Necesito llevar un control de los niveles



Inserto una marca al finalizar cada nivel

Resolución del Ejercicio 1

Seudocódigo Ejerc1-Niveles {

q: cola de vértices;

encolar raíz **R** en q;

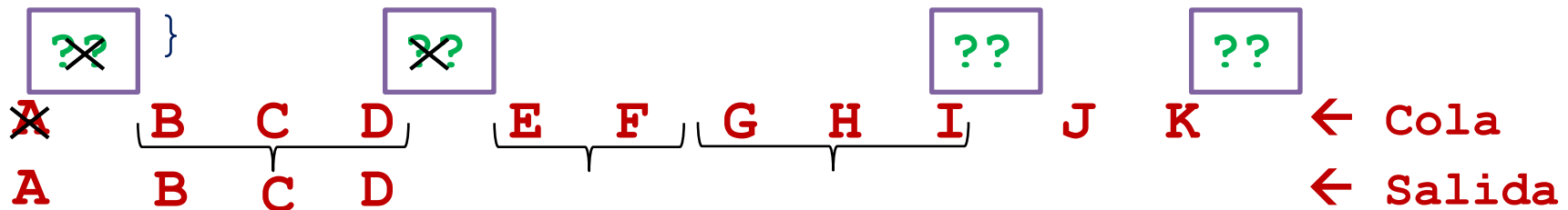
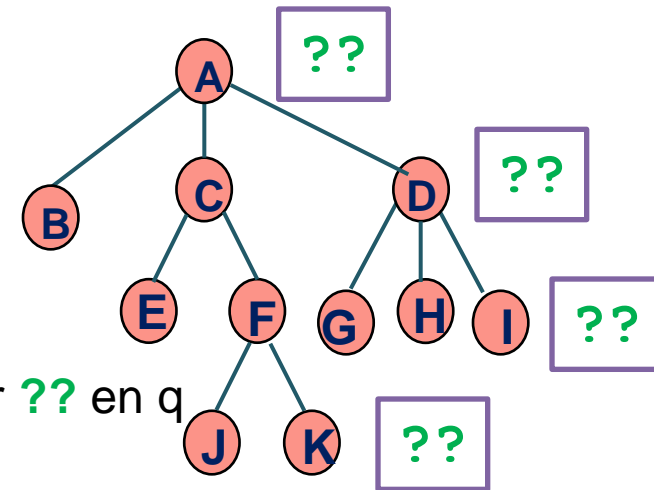
mientras (cola no se vacíe) {

desencolar **v** de q;

imprimir (dato de v);

para cada hijo **w** de **v**

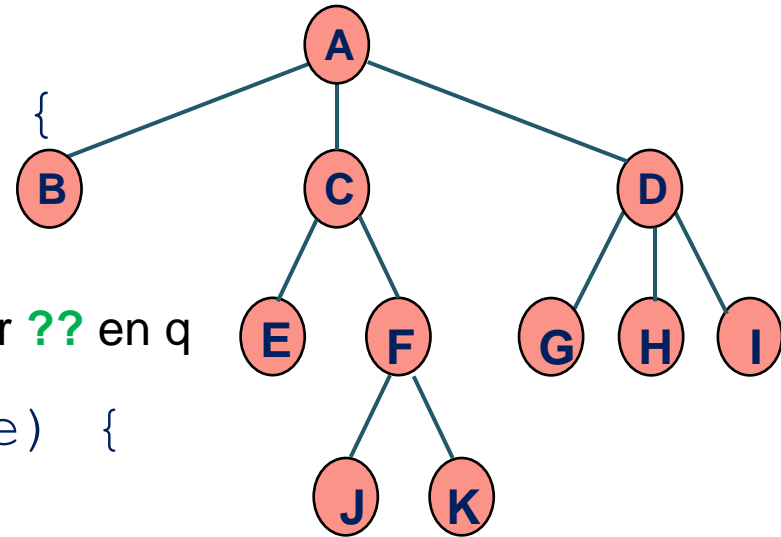
encolar **w** en q; }



Resolución del Ejercicio 1

Seudocódigo Ejerc1-Niveles

```
q: cola de vértices;  
encolar raíz R en q;  encolar ?? en q  
mientras (cola no se vacíe) {  
    desencolar v de q;  
    si (dato de v es ??) encolar ?? en q  
    sino  
        imprimir (dato de v);  
    para cada hijo w de v  
        encolar w en q; }  
}
```



Resolución del Ejercicio 1

Seudocódigo Ejerc1-Niveles {

```
q: cola de vértices;  
encolar raíz R en q; encolar ?? en q;  
mientras (cola no se vacíe) {  
    desencolar v de q;  
    si (dato de v es ?? ← encolar ?? en q  
        sino  
            si (dato de v no es ??)  
                imprimir (dato de v);  
            para cada hijo w de v  
                encolar w en q; }  
}
```



Resolución del Ejercicio 1

Seudocódigo Ejerc1-Niveles {

q: cola de vértices;

encolar raíz **R** en q; encolar ?? en q;

← cantNiveles = 0;

mientras (cola no se vacíe) {

desencolar **v** de q;

si (dato de **v** es ?? & q no está vacía) {

encolar ?? en q;

sino ← cantNiveles ++; }

si (dato de **v** no es ??) {

imprimir (dato de **v**);

para cada hijo **w** de **v**

encolar **w** en q; }

} ← return catNiveles++;

}







Resolución del Ejercicio 2

Seudocódigo Ejerc2-Niveles {

```
q: cola de vértices;           ← cantNodos: array de enteros;
encolar raíz R en q; encolar ?? en q;
mientras (cola no se vacíe) {   ← nroNivel = 0;
    desencolar v de q;
    si (dato de v es ?? & q no está vacía) {
        encolar ?? en q;       ← nroNivel ++; }
    sino
        si (dato de v no es ??) {
            imprimir (dato de v); ← cantNodos[nroNivel]++;
            para cada hijo w de v
                encolar w en q; }
        }
}
```

}

Resolución del Ejercicio 3

```
Seudocódigo Ejerc3-Niveles (int k) {  
    q: cola de vértices;  nroNivel = 0;  
    encolar raíz R en q; encolar ?? en q;  cantNodos = 0;  
    mientras (cola no se vacíe) {  
        desencolar v de q;  
        si (dato de v es ?? & q no está vacía) {  
            encolar ?? en q;  nroNivel ++ };  
        sino  
            si (dato de v no es ??) {  
                imprimir (dato de v);  
                si (nroNivel == k)   /*Contar la cantidad de  
                    cantNodos++;  
                para cada hijo w de v  
                    encolar w en q; }  
            }  
        }  
    }  
     return cantNodos;  
}
```

Resolución del Ejercicio 3

```
Seudocódigo Ejerc3-Niveles (int k) {  
    q: cola de vértices; nroNivel=0; cantNodos=0;  
    encolar raíz R en q; encolar ?? en q;  
    mientras (cola no se vacíe) {  
        desencolar v de q;  
        si (dato de v es ?? & q no está vacía){  
            encolar ?? en q;  
            nroNivel++;}  
        sino  
            si (dato de v no es ??)  
                si (nroNivel==k)  
                    mientras (dato de v no es ??)  
                        {cantNodos++;  
                        desencolar v de q;}  
                sino para cada hijo w de v  
                    encolar w en q;  
    }  
    return cantNodos;  
}
```