



Prueba Evaluación Continua Pilas-Colas

Algoritmos (Universidade da Coruña)

[Área personal](#) / [Cursos](#) / [UDC2021](#) / [614G010112021](#) / [Tema 2 / Lesson 2](#)
/ [Primera prueba de evaluación continua / First continuous evaluation test](#)

Comenzado el lunes, 26 de octubre de 2020, 13:09

Estado Finalizado

Finalizado en lunes, 26 de octubre de 2020, 13:15

Tiempo empleado 6 minutos 25 segundos

Calificación 10 de 10 (100%)

Pregunta **1**

Correcta

Puntúa 1 sobre 1

En una cola vacía indique el dato devuelto por `primero` tras realizar la siguiente secuencia de operaciones:

```
insertar a;  
insertar b;  
quitarPrimero;  
insertar c;  
primero
```

Responda sólo con una letra (sin espacios).

In an empty queue, indicate the data returned by `front` after the following operations:

```
enqueue a;  
enqueue b;  
dequeue;  
enqueue c;  
front;
```

Answer with only one letter (with no blanks).

Respuesta:

b



La respuesta correcta es: b

Pregunta **2**

Correcta

Puntúa 4 sobre 4

A partir de la siguiente declaración de tipos, y el siguiente procedimiento de creación de una lista:

tipo

```
Nodo = registro
    Elem: entero
    Sig: Pnodo
fin registro

Pnodo = puntero a nodo
Lista = pnodo
```

función CrearLista: Lista

```
CrearLista:= nil;
devolver CrearLista;
```

fin función

¿Cuáles de los siguientes procedimientos para buscar un elemento en la lista son correctos? (*Dar como correcta una opción no válida resta puntos*).

With the following type declaration and the following procedure to create a list:

type

```
Node = record
    Elem: integer
    Next: PNode
end record

PNode = pointer to a Node
List = PNode
```

function CreateList: List

```
CreateList = nil;
return CreateList;
```

end function

Which of the following procedures to search an element in the list are correct? (To mark a wrong answer as correct subtracts points).

Seleccione una o más de una:

☐ a. **función** buscar(x: entero, L: Lista): PNode

```
si L = nil entonces devolver L
sino
    buscar:= buscar(x, L^.Sig);
```

fin función

function search(x: integer, L: List): PNode

```
if L = nil then return L
else
    search:= search(x, L^.Next);
```

end function

☒ b. **función** buscar(x: entero, L:Lista): PNode

```

    tmp:= L;

    mientras tmp<>nil y tmp^.Elem<>x

    hacer

        tmp:= tmp^.Sig;

    fin mientras;

    devolver tmp;

fin función

```

```

function search(x: integer, L: List): PNode

    tmp = L;

    while tmp<>nil and tmp^.Elem<>x

    do

        tmp:= tmp^.Next;

    end while

    return tmp;

end function

```



```

función buscar(x: entero, L:
Lista): PNode
    tmp:= L;
    mientras tmp<>nil y
tmp^.Elem<>x
        hacer
            tmp:= tmp^.Sig;
        fin mientras;
    devolver tmp;

fin función

```

```

function search(x: integer,
L: List): PNode
    tmp = L;
    while tmp<>nil and
tmp^.Elem<>x
    do
        tmp:= tmp^.Next;
    end while;
    return tmp;
end function

```

☒ c. **función** buscar(x: entero, L: Lista): PNode

```

    mientras L<>nil y L^.Elem<>x

    hacer

        L:= L^.Sig;

    fin mientras;

    devolver L;

fin función

```

```

function search(x: integer, L: List): PNode

    while L<>nil and L^.Elem<>x

    do

        L:= L^.Next;

    end while

    return L;

end function

```



☐ d. **función** buscar(x: entero, L: Lista): Pnode

```

    mientras L^.Sig<>nil y L^.Sig^.Elem<>x

    hacer

        L:= L^.Sig;

    fin mientras

    devolver L^.Sig;

```

fin función

```
function search(x: integer, L: List): PNode
    while L^.Next<>nil and L^.Next^.Elem<>x
    do
        L := L^.Next;
    end while
    return L^.Next;
end function
```

Respuesta correcta

Las respuestas correctas son:

```
función buscar(x: entero, L:Lista): PNode
    tmp:= L;
    mientras tmp<>nil y tmp^.Elem<>x
    hacer

        tmp:= tmp^.Sig;
    fin mientras;
    devolver tmp;

fin función
```

```
function search(x: integer, L: List): PNode
    tmp = L;
    while tmp<>nil and tmp^.Elem<>x
    do
        tmp:= tmp^.Next;
    end while
    return tmp;
end function
,
```

```
función buscar(x: entero, L: Lista): PNode
    mientras L<>nil y L^.Elem<>x
    hacer
        L:= L^.Sig;
    fin mientras;
    devolver L;

fin función
```

```
function search(x: integer, L: List): PNode
    while L<>nil and L^.Elem<>x
    do
        L:= L^.Next;
    end while
    return L;
```

```
end function
```

Pregunta **3**

Correcta

Puntúa 1 sobre 1

En una pila vacía, indique el dato devuelto por `cima` tras la siguiente secuencia de operaciones:

```
apilar a;  
apilar b;  
desapilar;  
apilar c;  
cima
```

Responda sólo con una letra (sin espacios).

In an empty stack, indicate the data returned by `top` after the following operations:

```
push a;  
push b;  
pop;  
push c;  
top;
```

Answer with only one letter (with no blanks).

Respuesta:



La respuesta correcta es: c

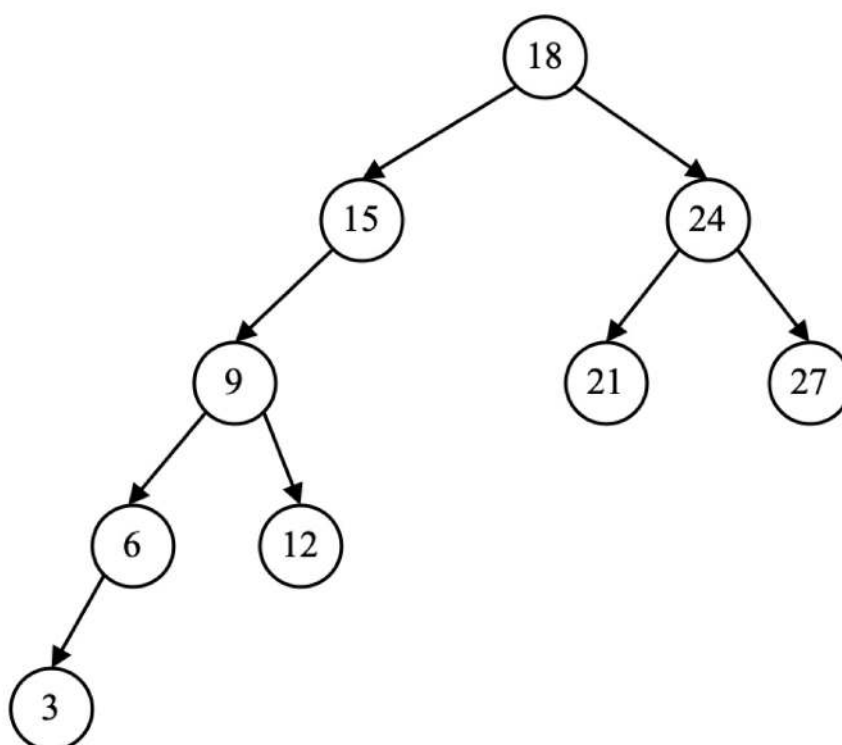
Pregunta 4

Correcta

Puntúa 4 sobre 4

¿Cómo quedaría el siguiente árbol binario de búsqueda si eliminásemos, en el orden que sigue, las claves 24 y 18? (Dar como correcta una opción no válida resta puntos)

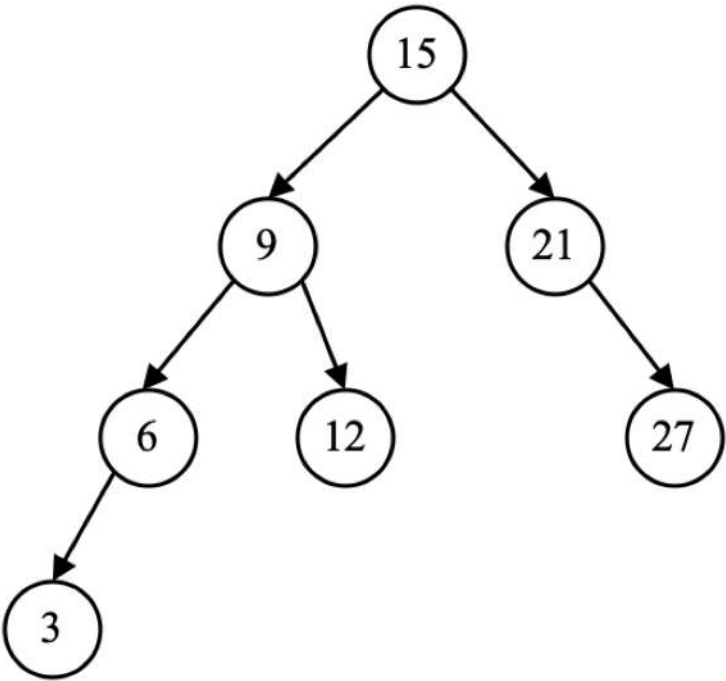
What would the following Binary Search Tree look like if we delete, in the order that follows, the keys 24 and 18? (To give a non-valid answer as correct subtracts points)



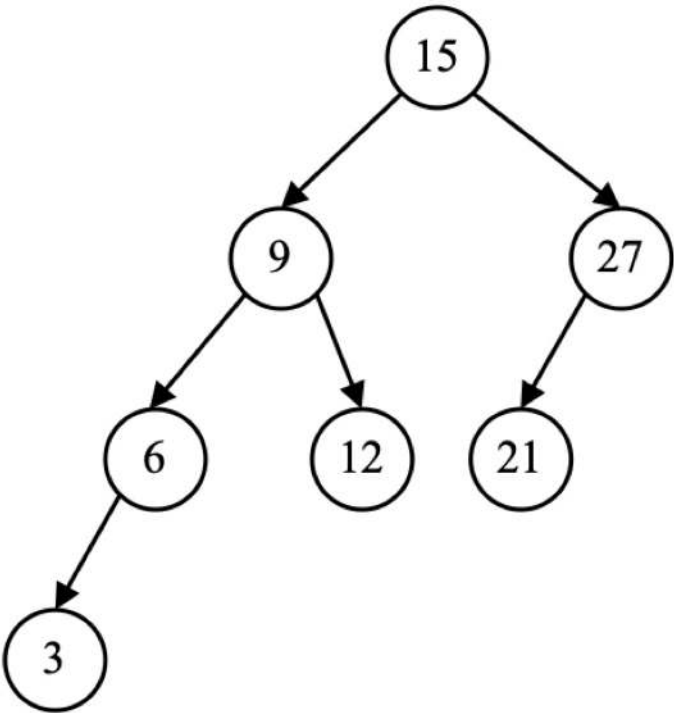
Seleccione una o más de una:

☒ a.



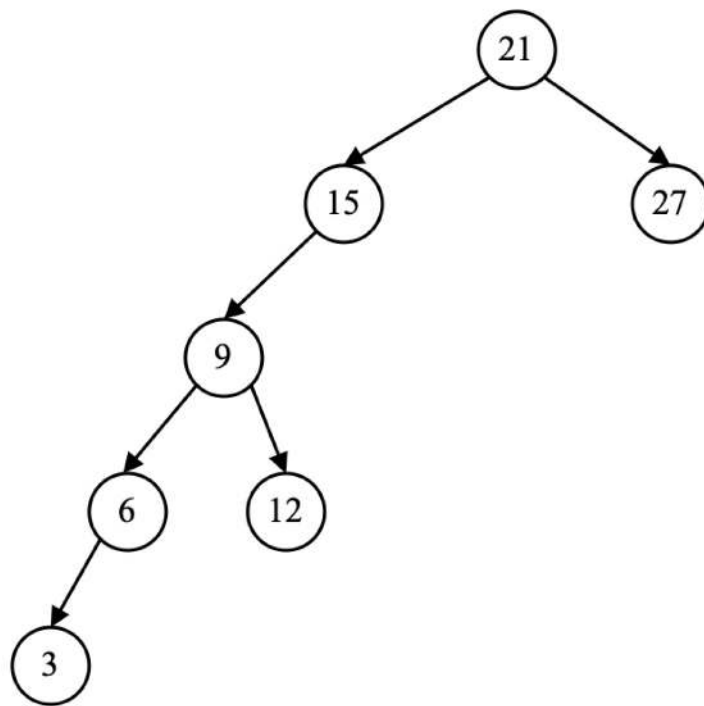


☐ b.

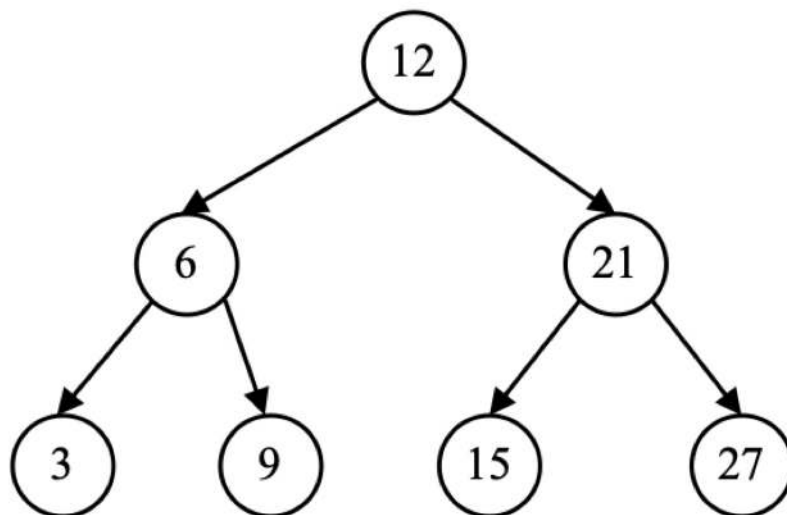


☒ c.





☐ d.



Respuesta correcta

Las respuestas correctas son:

