UNIVERSIDAD DE OVIEDO
Departamento de Informática

ASIGNATURA				CENTRO / TITULACION				
APELLIDOS Y NOMRE: Iglesias Ramos, Hernán								CALIFICACION
DNI / ID				FECHA		CURSO	GRUPO	NUMERO	HOJA

(1)

```
public class Fibonacci implements Iterable<E> (){          // Integer
    private int n;
    public Fibonacci (int n){
        this.n = n;
    }
@Override
public <E> iterator (){
    return new FibIterator();
}
private class FibIterator implements Iterator<E> {          // Integer
    int term = 1;
    int previous = 0;
    int k = 0;
    public boolean hasNext(){
        return k < n;
    }
    public Integer next (){
        if(!hasNext()){
            if(n == 0){                                     // k
                return previous;                            // k++
            }
            if(n == 1){                                     // k
                return term;                                // k++
            }
            throw NoSuchElementException();
        }
        while (k <= n){
            term += previous;
            previous = term - previous;
            k++;
        }
    return term;
}
```

2'5/10

¿?

(1)

2.

```
public static void print (int n) {
    Fibonacci fib = new Fibonacci (n);
    for (int x : fib) {
        System.out.printf ("%d", x);
    }
    System.out.println ();
}
```

10/10

term=1 prev=0 k=2	term=1 prev=1 k=3	term=2 prev=1 k=4	term=3 prev=2 k=5	term=5 prev=3 k=6	term=8 prev=5 k=7	term=13 prev=8 k=8	term=21 prev=13 k=9	term=34 prev=21 k=10	term=55 prev=34 k=11
n=10	return 1	return 2	return 3	return 5	return 8	return 13	return 21	return 34	return 55

1,2,3,5,8,13,21,34,55

UNIVERSIDAD DE OVIEDO
Departamento de Informática

ASIGNATURA				CENTRO / TITULACION					
APELLIDOS Y NOMBRE: IGLESIAS RAMOS, HERNÁN									CALIFICACION
DNI / ID				FECHA		CURSO	GRUPO	NUMERO	HOJA

(2) FORMA SI HUBIERA CONSTRUCTOR POR DEFECTO // SIN CONSTRUCTOR POR DEFECTO

1.

```
public C ( Collection<? extends E> c ){
    this();
    data.addAll(c);
}
```

repetidos y nulos

4/10

⇒ ok

```
public C (Collection<? extends E> c){
    data = new LinkedList<>();
    data.addAll(c);
}
```

```
public boolean add (E e){
    if( data.contains(e) ){          // SI ESTÁ CONTENIDO NO SE AÑADE
        return false;
    }
    else{
        data.add(e);
        return true;
    }
```

2.

```
public C (Collection<? extends E> c){
    if( c instanceof C ){
        C<E> other = (C<E>) c;
        this.data = other.data;          // PARA COMPARTIR LA REPRESENTACIÓN
    }
    else{
        this.data = new LinkedList<>(c);
    }
}
```

6/10

repetidos y nulos

UNIVERSIDAD DE OVIEDO
Departamento de Informática

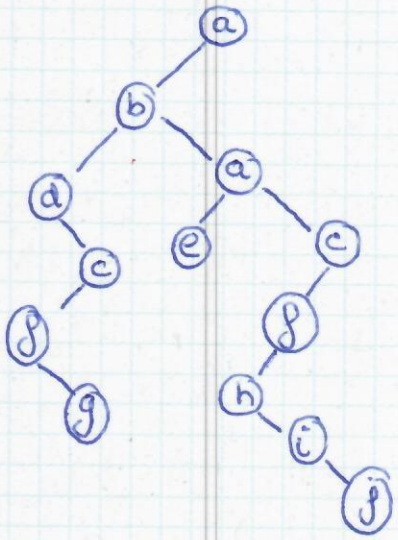
ASIGNATURA			CENTRO / TITULACION				
APELLIDOS Y NOMRE: IGLESIAS RANCS HERNÁN							CALIFICACION
DNI / ID		FECHA		CURSO	GRUPO	NUMERO	HOJA

(3)

1. PostOrden : sub-árbol izq, sub-árbol der, raíz

(d, f, g, c, b, e, a, h, i, f, f, c, a) 10/10

2. Árbol binario

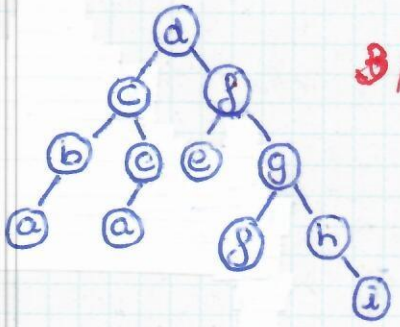


7/10

Inorden: sub-árbol izq, raíz, subárbol der

(d, f, g, c, b, a, e, a, h, i, f, f, c).

3.



8/10