

Algoritmos y Programación 1

Apellido y Nombres:

DNI:

E1	E2	E3	E4	E5	Calificación

Ejercicio 1 { java, arrays }

Implementar según la siguiente especificación, agregando pre y post-condiciones.

Incorporar alguna validación de dominio.

Implementar la clase **Estacionamiento** tal que

1. Se construya a partir de la cantidad máxima de autos que pueden ser alojados.
2. Agregue un auto, indicando su patente y cuánto duró la estadía (en minutos).
3. Devuelva la patente de un auto a partir de la posición.
4. Cuente la cantidad de autos del día
5. Calcule el dinero total que ingresó, asumiendo que se cobra \$2 por minuto por auto.

```
class Estacionamiento {  
    public Estacionamiento(int) { /* ... */ }  
    public void ingresarAuto(String, int) { /* ... */ }  
    public String obtenerPatente(int) { /* ... */ }  
    public int cantidadDeAutos() { /* ... */ }  
    public int dineroRecaudado() { /* ... */ }  
}
```

Ejercicio 2 { pruebas }

Plantear cinco (5) pruebas para el problema del punto anterior. Los casos planteados deberán ser **conceptualmente** diferentes. Escribir el código en JUnit para tres (3) de esos casos.

Algoritmos y Programación 1

Apellido y Nombres:

DNI:

Ejercicio 3 { memoria }

Realizar los diagramas de Stack y Heap para los momentos #uno y #dos del siguiente fragmento de código.

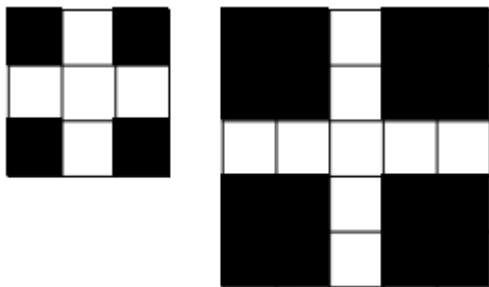
```
Equipo equipo = new Equipo(10);
Miembro gandalf = new Miembro("Sam", "Hobbit");
Miembro frodo = new Miembro("Legolas", "Elfo");
Miembro pippin = frodo;
frodo = gandalf;
// #uno
gandalf = null;
equipo.agregar(frodo, 8);
equipo.agregar(pippin, 10);
equipo.agregar(new Miembro("Gimli", "Enano"), 8);
frodo = null;
Assert.assertEquals(equipo.obtenerMiembro(10), pippin);
// #dos
```

Ejercicio 4 { algoritmos, matrices }

Dada una matriz de **Color** y de tamaño **n x n**, siendo **n** impar y mayor o igual a 3, y un **Color** determinado, escribir un algoritmo para pintar un lienzo que originalmente es de **Color.BLANCO** con el patrón solicitado. Se debe **pintar sobre el lienzo original**. No se recomienda pintar de **Color.BLANCO**.

```
public void pintarLienzo(Color[][] lienzo, Color pincel) {
    // tu código aquí
}
```

Se proporcionan ejemplos para $n = 3$ y $n = 5$



Ejercicio 5 { lógica }

Explique y ejemplifique con código Java el concepto de conjunción binaria (AND). Escriba la tabla de verdad para dicho concepto. El ejemplo deberá ser aplicado y no conceptual (las variables deben tener sentido).