

Sesión 07: Invocación de métodos

Hoja de problemas

Programación 2

Ángel Herranz

aherranz@fi.upm.es

Universidad Politécnica de Madrid

Febrero 2019

- ☐ **Ejercicio 1.** Continuando con la modelización del juego de cartas *Texas hold'em*, implementa una nueva clase *Mano* que represente las dos cartas de un jugador.
- ☐ **Ejercicio 2.** Ahora te toca implementar una nueva clase *Mano* que represente las dos cartas de un jugador.
- ☐ **Ejercicio 3.** Si no se te ha ocurrido, puedes hacer que el constructor reparta cartas aleatoriamente.
- ☐ **Ejercicio 4.** Implementa una nueva clase *Comunitarias* que represente las cinco cartas comunitarias.
- ☐ **Ejercicio 5.** ¿Has representado las cuatro fases?
Flop tres cartas descubiertas.
Turn cuatro cartas descubiertas.
River las cinco cartas descubiertas.
- ☐ **Ejercicio 6.** Escribe una *función* *mejorJugada* en el programar principal que reciba unas cartas comunitarias y dos manos y decida qué mano es la ganadora. Úsala en un programa principal. Algo como esto:


```
public class Compara {  
    /**  
     * Devuelve 1 si m1 es ganadora, -1 si m2 es ganadora y 0 si son iguales  
     */  
    private static int mejorJugada(Comunitarias c,  
                                    Mano m1,  
                                    Mano m2) {  
        ...  
    }  
}
```


```

    }

    public static void main(String args) {
        Mano m1 = new Mano();
        Mano m2 = new Mano();
        Comunitarias c = new Comunitarias();
        switch (mejorJugada(c, m1, m2) {
            case 0:
                System.out(m1 + " es igual jugada que " + m2);
                break;
            case 1:
                System.out(m1 + " es mejor jugada que " + m2);
                break;
            case -1:
                System.out(m1 + " es peor jugada que " + m2);
                break;
        }
    }
}

```

 **Ejercicio 7.** Hoy en clase, ha surgido múltiples preguntas alrededor del significado del modificador **static**...

 **Ejercicio 8.** Ordenar las canciones de mayor a menor valoración