



Programació Avançada

Tècnica Backtracking

Cartes set i mig



Enunciat

Tècnica: Backtracking

- Al joc del **set i mig** s'hi juga amb les cartes de la baralla espanyola: **1,2,3,4,5,6,7,10,11 i 12** dels quatre pals (**Oros, Copes, Espases, Bastons**). Es vol escriure un programa per obtenir 7.5, fent valer cada carta pel seu número de carta, excepte els 10, 11 i 12 que tenen un valor de 0.5. L'algorisme ha de trobar i visualitzar per **pantalla totes les combinacions possibles que facin 7,5**.
- Per cada carta de cada solució cal visualitzar: nom i pal. **Per exemple:**
7 de Bastos – Rei de Copes
/*és una solució al problema */



Tècnica: Backtracking

```
public class Carta{
    private String nom; /*as, rei, cavall, ...*/
    private String pal; /*Oros, Copes, Espases,
                        Bastons */
    private float valor; /*valor que té la carta
                        en el joc */
    public Carta(String n, String pa, int val){
        nom=n; pal=pa; valor=val;
    }
    public String getPal(){return pal;}
    public String getNom(){ return nom;}
    public float getValor(){ return valor;}
}
```



Tècnica: Backtracking

```
public class Solucio{
    private Carta []cartes;
    //dades totes les cartes de la baralla espanyola
    private Carta []solucio; //magatzem per construir la
                             // solució

    public Solucio(){
        //sentències que creen i emplenen el magatzem amb
        // totes les cartes i es crea i inicialitza a nuls
        // el magatzem solució
    }

    public static void main(String args[]){
        //sentències
    }

    public ??? Backtracking(????????????){
        //denoteu que aquest procediment NO és estàtic
        //sentències
    } //fi procediment
} // fi classe
```



Tècnica: Backtracking

- **Exercici 2 – Modificació de l'anterior.**

Repetir l'exercici anterior, ara es volen trobar, només, **cinc solucions** al problema. L'algorisme seguirà el mateix espai de cerca?

- **Exercici 3 – Modificació de l'anterior.**

Repetir l'exercici anterior, ara es vol trobar la millor solució. La millor solució és aquella que **té més cartes**.

- **Exercici 4 – Modificació de l'anterior.**

Repetir l'exercici anterior, ara es volen trobar totes les solucions, a l'igual que el primer exercici però les solucions **no han de tenir cartes del mateix pal**.

- **Exercici 5.**

Implementeu el mètode constructor de la classe Solucio. **Pendent**