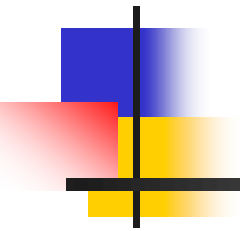

Programació Avançada

Tècniques de disseny d'algorismes – Backtracking

Factories

Tècnica de backtracking

Factories

- **Enunciat**: 10 factories de tecnologia punta necessiten una instal·lació informàtica i personal per dirigir-les. El govern de l'estat els ha assignat 10 màquines i 10 tècnics. No totes les màquines són escaients per les necessitats de totes les empreses, no tots els tècnics coneixen totes les màquines ni tots els tècnics accepten treballar amb totes les empreses. Per això es disposa de funcions **booleanes**:
  - `adequada(m,e)`
  - `coneix(t,m)`
  - `accepta(t,e)`

Tenim restriccions

- Dissenyar un algorisme per trobar **una manera** d'assignar a cada empresa una màquina escaient per a les seves necessitats i un tècnic que la conegui i accepti treballar a l'empresa.

Tècnica de backtracking

Factories

- Suposem l'existència de les classes: Empresa, Maquina i Tecnic (totes amb el mètode toString() redefinit).

```
public class Factories{
        private Empresa e[];
        private Tecnic t[];
        private Maquina m[];
        private class Tripletes{
                private Empresa e;
                private Tecnic t;
                private Maquina m;
        }
        private Tripletes sol[];
```

Tècnica de backtracking

Factories

```
public Factories(){ //constructor
   e=Empresa.carregarFitxerE();
   m=Maquina.carregarFitxerM();
   t=Tecnic.carregarFitxerT();
```

Suposem la seva existència

```
   sol=new Tripletes[10];
    for (int i=0; i<10; i++){
          sol[i]=new Tripletes();
          sol[i].e=e[i];
   }
}
```

Creem el magatzem

```
public static boolean adequada(Maquina m, Empresa z)
{ /* sentències*/ }
public static boolean coneix(Tecnic t, Maquina z)
{/* sentències*/ }
public static boolean accepta(Tecnic t, Empresa z)
{ /* sentències*/ }
```

Tècnica de backtracking

Factories

- Anàlisi
  - S'han de prendre decisions. A cada empresa cal trobar-li una màquina i un tècnic (E,M,T)
  - Plantejament: fer 2 backtrackings
    - **Bactracking 1**: assignar a cada empresa una màquina. Quan tenim 10 parelles Empresa-Màquina correctes (una solució) engeguem el segon backtracking.
    - **Backtracking 2**: a cada parella (E,M) li assigno un tècnic. Si amb les parelles que arriben no podem trobar solució demanar al backtracking 1 una altra solució!!!

Tècnica de backtracking

Factories

```
public static void main (String args[]){
      Factories f = new Factories();
      boolean marcats[] = new boolean[10];
      for (int i=0; i<10; i++)
        marcats[i]=false;
      if (f.back1(0,marcats))
        for (int i=0; i<10; i++)
          System.out.println(f.sol[i].e.toString()+
                             f.sol[i].t.toString()+
                            f.sol[i].m.toString());
      else
        System.out.println("NO existeix solució");
}
```

Tècnica de backtra

Factories

```
public boolean back1(int k,boolean
   boolean trobat =false;
   for (int i=0; i<10 && !trobat; i++){
       if (!marcats[i] && adequada(m[i],sol[k].e)){
             sol[k].m=m[i]; marcats[i]=true;
             if (k==9){  //solucio
                   boolean []marc=new boolean[10];
                   for (int j=0; j<10; j++)
                         marc[j]=false;
                   trobat=back2(0,marc);
             }
             else   trobat=back1(k+1,marcats);
             if (!trobat){ marcats[i]=false;}
       } // fi acceptable
   } //fi for
   return trobat;
}
```

i→Màquina
K→ Empresa

```
public static boolean Back1Solucio( TaulaSolucio  TS, int k){
   boolean trobada=false;
   inicialitzem_valors_domini_decisio_k
   agafar_el_primer_valor
   while (quedin_valors && !trobada){
       if (valor_acceptable){  //no viola les restriccions
             anotem_el_valor_a_la_solucio
             if (solucio_final) trobada=true;
             else if (solucio_completable)
                   trobada=Back1Solucio(TS, k+1);
             if (!trobada) desanotem_el_valor
       } //fi if
       agafar_seguent_valor
        //passem al següent germà a la dreta
   } // fi while
   return trobada;
} // fi procediment
```

Esquema de cerca

Important

Opcionalment es pot fer tractament abans d'aca el procedimer

Tècnica de backtracking

Factories

```
public boolean back2(int k,boolean[]marcats){
   boolean trobat =false;
   for (int i=0; i<10 && !trobat; i++){
       if (!marcats[i] && coneix(t[i], sol[k].m) &&
                                  accepta(t[i],sol[k].e)){
                sol[k].t=t[i]; marcats[i]=true;
                if (k==9) trobat=true;
                else trobat=back2(k+1,marcats);
                if (!trobat){ marcats[i]=false;}
       } // fi acceptable
   } //fi for
   return trobat;
}
```

$k \rightarrow E_i, M_i$

i→ tècnic