

# TP3 Graphe et Optimisation

Arthur Delain

## Part I

# Les programmes

## 1 PDS

```
int n=...;
range Nodes= 1..n;

tuple Edge{ int u; int v;}
{Edge} Edges with u in Nodes, v in Nodes=...;
{int} NbsIn[i in Nodes]={j|<i,j> in Edges};
{int} NbsOut[i in Nodes]={j|<j,i> in Edges};

dvar boolean x [Nodes];

minimize
sum(i in Nodes) x[i];

subject to{
    forall(i in Nodes)
x[i] + sum(j in NbsIn[i]) x[j] + sum(j in NbsOut[i]) x[j] >=1;
};

{int} Dom={i|i in Nodes : x[i]==1};

execute
{
writeln("Dominant Sommets",Dom);
}
```

## 2 PIS

```
int n=...;
range Nodes= 1..n;

tuple Edge { int u; int v; }
{Edge} Edges with u in Nodes, v in Nodes=...;

dvar boolean x [Nodes];

maximize
    sum(i in Nodes) x[i];

subject to{
    forall (<u,v> in Edges)
        x[u] + x[v] <= 1;
};

{int} Stb={i|i in Nodes : x[i]==1};

execute
{
    writeln("Ensemble Stable",Stb);
}
```

### 3 PMIS

```
int n=...;
range Nodes= 1..n;

tuple Edge { int u; int v;}

{Edge} Edges with u in Nodes, v in Nodes=...;
{int} NbsIn[i in Nodes]={j|<i,j> in Edges};
{int} NbsOut[i in Nodes]={j|<j,i> in Edges};

dvar boolean x [Nodes];

minimize
sum(i in Nodes) x[i];

subject to{
  forall(i in Nodes)
    x[i] + sum(j in NbsIn[i]) x[j] + sum(j in NbsOut[i]) x[j] >=1;
  forall (<u,v> in Edges)
    x[u] + x[v] <= 1;
};

{int} Dom={i|i in Nodes : x[i]==1};

execute
{
writeln("Ensemble Stable",Dom);
}
```

## Part II

# Les resultats

### 4 PDS

```
//Exemple1
n=10;
Edges = {<1,2>, <1,5>, <1,6>, <1,7>, <2,3>, <3,4>, <4,8>, <4,9>, <4,10>};
// solution (optimal) with objective 2
Dominant Sommets {1 4}

//Exemple2
n=11;
Edges = {<1,2>, <1,3>, <1,4>, <2,3>, <3,5>, <3,6>,
<4,5>, <4,7>, <5,6>, <5,8>, <7,8>, <7,9>, <8,10>, <9,10>, <10,11>};
// solution (optimal) with objective 3
Dominant Sommets {3 7 10}

//Exemple3
n=17;
Edges = {<1,2>, <1,3>, <1,4>, <1,5>, <2,3>, <2,6>, <3,4>, <3,7>,
<4,5>, <4,8>, <5,9>, <6,7>, <6,10>, <7,8>, <7,11>, <8,9>, <8,12>, <9,13>,
<10,11>, <10,14>, <11,12>, <11,15>, <12,13>, <12,16>, <13,17>, <14,15>, <15,16>, <16,17>};
// solution (optimal) with objective 4
Dominant Sommets {3 9 10 16}
```

## 5 PIS

```
//Exemple1
n=10;
Edges ={<1,2>,<1,5>,<1,6>,<1,7>,<2,3>,<3,4>,<4,8>,<4,9>,<4,10>};
// solution (optimal) with objective 7
Ensemble Stable {2 5 6 7 8 9 10}

//Exemple2
n=11;
Edges ={<1,2>,<1,3>,<1,4>,<2,3>,<3,5>,<3,6>,<4,5>,<4,7>,<5,6>,<5,8>,<7,8>,<7,9>,<8,10>,<9,10>,<10,11>};
// solution (optimal) with objective 6
Ensemble Stable {2 4 6 8 9 11}

//Exemple3
n=17;
Edges ={<1,2>,<1,3>,<1,4>,<1,5>,<2,3>,<2,6>,<3,4>,<3,7>,<4,5>,<4,8>,<5,9>,<6,7>,<6,10>,<7,8>,<7,11>,<8,9>,<8,12>,<9,13>,<10,11>,<10,14>,<11,12>,<11,15>,<12,13>,<12,16>,<13,17>,<14,15>,<15,16>,<16,17>};
// solution (optimal) with objective 8
Ensemble Stable {2 4 7 9 10 12 15 17}
```

## 6 PMIS

```
//Exemple1
n=10;
Edges ={<1,2>,<1,5>,<1,6>,<1,7>,<2,3>,<3,4>,<4,8>,<4,9>,<4,10>};
// solution (optimal) with objective 2
Ensemble Stable {1 4}

//Exemple2
n=11;
Edges ={<1,2>,<1,3>,<1,4>,<2,3>,<3,5>,<3,6>,<4,5>,<4,7>,<5,6>,<5,8>,<7,8>,<7,9>,<8,10>,<9,10>,<10,11>};
// solution (optimal) with objective 3
Ensemble Stable {3 7 10}

//Exemple3
n=17;
Edges ={<1,2>,<1,3>,<1,4>,<1,5>,<2,3>,<2,6>,<3,4>,<3,7>,<4,5>,<4,8>,<5,9>,<6,7>,<6,10>,<7,8>,<7,11>,<8,9>,<8,12>,<9,13>,<10,11>,<10,14>,<11,12>,<11,15>,<12,13>,<12,16>,<13,17>,<14,15>,<15,16>,<16,17>};
// solution (optimal) with objective 4
Ensemble Stable {4 6 13 15}
```