

antenne.dat

points = {

<1,1>,

<1,2>,

<1,3>,

<1,4>,

<1,5>,

<2,1>,

<2,2>,

<2,3>,

<2,4>,

<2,5>,

};

antennes = {

<1,100>,

<1,200>,

};

visibilite = [

[1 1 1 0 0 0 0 0 0 0]

[1 1 1 0 0 0 0 0 1 0]

[1 1 1 0 0 0 1 0 0 1]

[0 1 1 1 0 0 1 0 1 0]

[1 1 1 0 1 1 0 1 1 1]

[1 1 1 0 1 1 0 0 0 1]

[1 0 1 1 0 0 1 1 1 1]

[1 1 1 0 0 0 1 1 1 0]

[1 0 1 1 0 0 0 1 1 1]

[1 1 1 0 0 0 1 0 1 1]

];

antenne.mod

```
tuple tuplepoint{
int x;
int y;
};
{ tuplepoint } points = ...;

tuple tupleantenne{
int portee;
int prix;
};
{ tupleantenne } antennes = ...;
float dx;
float dy;
int visibilite[points][points]=...;

float Distances[points][points];

int select[points][antennes];

execute{
  for(var p1 in points){
    for(var p2 in points){
      dx = p1.x - p2.x;
      dy = p1.y - p2.y;
      Distances[p1][p2]= Opl.sqrt(dx*dx+dy*dy);
    }
  }
}

}
dvar boolean SELECT[antennes][points];

// minimiser le cout total des antennes choisies
minimize sum(a in antennes, p in points) a.prix*SELECT[a][p];

subject to {
  // pour chaque point, il faut au moins que une des antennes qui permet de le couvrir
  // soit allumee.
  // filtre pour lister les antennes qui couvre un point:
  // - le point pa ou est l'antenne est visible a partir du point p a couvrir
  // - ET la distance entre p et pa est inferieur a la portee de l'antenne a.portee
  forall(pAcouvrir in points)
    sum(a in antennes, pa in points: (visibilite[pAcouvrir][pa] == 1)
```

```

                                && (Distances[pAcouvrir][pa] <= a.portee))
SELECT[a][pa] >= 1;

}

execute {
    // affichage du resultat apres optimisation de SELECT
    for(var a in antennes) for(var p in points) if (SELECT[a][p] == 1)
        writeln( a.portee, " : ", p.x, "@", p.y);
    writeln(SELECT);
}

/*
subject to {
int sum=0;
int total=0;
    forall(p in points){
        for(a in antennes){
            for(p2 in points){
                if(select[p2][a]==1){
                    if(distance[p][p2]<=a.portee && visibilite[p][p2]){
                        sum=sum+1;
                        total=total+a.prix;
                    }
                }
            }
        }
        sum>=1;
    }
}

}*/

```