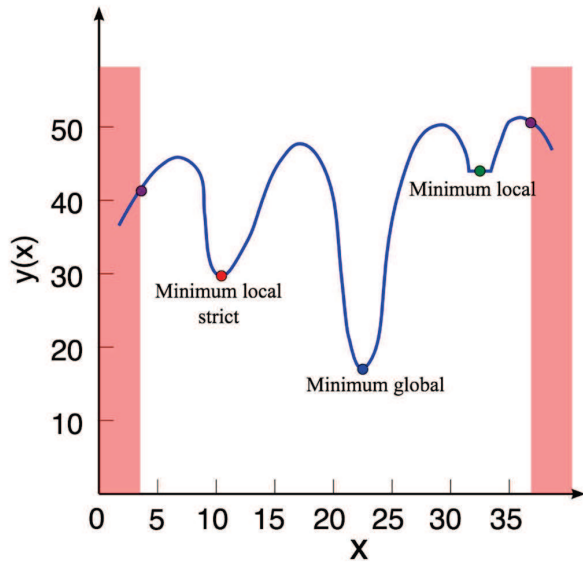


# Optimisation par algorithmes gloutons

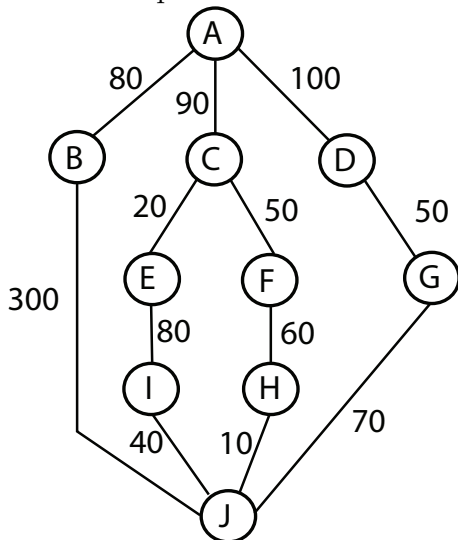
## 1 Notion d'optimisation



## 2 Algorithmes gloutons : principe

### 2.1 Exemple : graphe de distances entre des villes

Chemin le plus court entre A et J :



## 2.2 Résolution par algorithme glouton

La résolution du problème d'optimisation se fait par une décomposition en sous-problèmes beaucoup plus simples à résoudre. La concaténation (l'assemblage) des différentes solutions des sous-problèmes constitue alors la solution **supposée** au problème initial.

- Un problème est décomposé en une succession de problèmes plus simple.
- Chacun est résolu après l'autre.
- Pour chacun, l'algorithme choisit la solution localement la meilleure.
- Cette solution est utilisée comme donnée pour la résolution du problème suivant.

## 2.3 Remarques

- Attention, ce principe n'assure absolument pas de trouver la « vraie » solution optimale du problème mais fournit une solution généralement intéressante et valide.
- Certains problèmes bien conditionnés amène une solution gloutonne optimale.

## 3 Application : occupation de la salle informatique

Plusieurs professeurs souhaitent utiliser la nouvelle salle informatique. Leur emploi du temps respectif est donné ci-dessous. On cherche à optimiser l'occupation de la salle.

Proposer la solution gloutonne aux problèmes d'optimisation :

1. occupation maximale, en commençant par 8h ;
  - On choisit le professeur commençant le plus tôt, et restant le plus longtemps dans la salle ;
  - Le premier choix étant fait, on connaît l'horaire à laquelle se libère la salle. On peut recommencer avec ce nouvel horaire.
2. occupation maximale, en commençant par 18h ;
3. Nombre d'utilisateurs maximal.

