Les algorithmes

Gérard Berry

Chaire d'innovation technologique Liliane Bettencourt

Collège de France

25 janvier 2008

La science informatique

- Théorie de l'information : coder et transporter efficacement
- Algorithmique : faire vite et bien
- Théorie de la programmation : écrire vite et juste
- Reliées : automatique & signal, analyse numérique

Art d'organiser un calcul complexe
en partant d'opérations simples

La science informatique

- Théorie de l'information : coder et transporter efficacement
- Algorithmique : faire vite et bien
- Théorie de la programmation : écrire vite et juste
- Reliées : automatique & signal, analyse numérique

Art d'organiser un calcul complexe
en partant d'opérations simples

Euclide, Archimède, Brahmagupta, Al Khuwārizmī, Fibonacci
Hilbert, Turing, Church, Gödel, Von Neumann, Knuth, Karp,...

Al Khuwārizmī
~ 783 - 850

algorithme
algèbre

Source Jean Vuillemin

Les domaines algorithmiques

texture éclairage
fusion mise 3D
contours segmentation
corrections optiques
images

trajectoires
déformations
tomographie
surfaces volumes
géométrie

gestion de trafic
diffusion
protocoles
codage
réseaux

éléments finis
matrices prog. linéaire
nb. premiers cryptage
4 opérations
nombres

grammaires
automates
croisement
tri, recherche
classement
mots, textes

emploi du temps
circulation
routage
optimisation

Les critères algorithmiques

- La machine: séquentielle, parallèle, distribuée?
- La correction
- Le temps de calcul (pire ou moyen)
- La taille mémoire (pire ou moyenne)
- La consommation d'énergie
- Exact ou approché? Prévisible ou imprévisible? ...

Les critères algorithmiques

- La machine: séquentielle, parallèle, distribuée?
- La correction
- Le temps de calcul (pire ou moyen)
- La taille mémoire (pire ou moyenne)
- La consommation d'énergie
- Exact ou approché? Prévisible ou imprévisible? ...

Des milliers de compromis pour des milliers de problèmes
Comment choisir ? Analyse mathématique sophistiquée !

Les machines

- Machine séquentielle de von Neumann
  - mémoire + unités de calcul
  - opérations séquentielles : lire, écrire, additionner, ...
- Machines parallèles asynchrones
  - communication par courrier sur réseau
- Machines parallèles synchrones
  - communication conceptuellement instantanée (p.e., circuits)
- Machines mixtes
  - System on Chip

Les machines

- Machine séquentielle de von Neumann
  - mémoire + unités de calcul
  - opérations séquentielles : lire, écrire, additionner, ...
- Machines parallèles asynchrones
  - communication par courrier sur réseau
- Machines parallèles synchrones
  - communication conceptuellement instantanée (p.e., circuits)
- Machines mixtes
  - System on Chip

Rendez-vous vers 13h au pied du téléski
- autrefois infernal (il n'est pas là, j'en fait encore un)
- maintenant trivial avec le portable

Les structures algorithmiques

objets	bits, entiers, flottants graphes, matrices mots, images
structures de données	piles, listes, arbres, tables,... tableaux, chaînages diagrammes de Voronoi
structures de contrôle	séquence, boucle récursion parallélisme synchrone parallélisme aynchrone

Principes : diviser pour régner, exploiter l'aléa, etc.

Cinq exemples

- Le tri
- L'addition
- L'enveloppe convexe
- Les diagrammes de Voronoi
- Le problème SAT
- Plus une friandise....

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11

39 45 14 18 15 31 91 24

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ 10 11 12 17 23 33 77 83

39 45 14 18 15 31 91 24 ⟶ 14 15 18 24 31 39 45 91

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ 10 11 12 17 23 33 77 83

39 45 14 18 15 31 91 24 ⟶ 14 15 18 24 31 39 45 91

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10~~ 11 12 17 23 33 77 83

39 45 14 18 15 31 91 24 → 14 15 18 24 31 39 45 91

10

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11~~ 12 17 23 33 77 83

39 45 14 18 15 31 91 24 → 14 15 18 24 31 39 45 91

---

10 11

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12~~ 17 23 33 77 83

39 45 14 18 15 31 91 24 → 14 15 18 24 31 39 45 91

---

10 11 12

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12~~ 17 23 33 77 83

39 45 14 18 15 31 91 24 → ~~14~~ 15 18 24 31 39 45 91

---

10 11 12 14

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12~~ 17 23 33 77 83

39 45 14 18 15 31 91 24 → ~~14 15~~ 18 24 31 39 45 91

---

10 11 12 14 15

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17~~ 23 33 77 83

39 45 14 18 15 31 91 24 → ~~14 15~~ 18 24 31 39 45 91

---

10 11 12 14 15 17

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17~~ 23 33 77 83

39 45 14 18 15 31 91 24 → ~~14 15 18~~ 24 31 39 45 91

---

10 11 12 14 15 17 18

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17 23~~ 33 77 83

39 45 14 18 15 31 91 24 → ~~14 15 18~~ 24 31 39 45 91

---

10 11 12 14 15 17 18 23

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23~~ 33 77 83

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24~~ 31 39 45 91

---

10 11 12 14 15 17 18 23 24

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17 23~~ 33 77 83

39 45 14 18 15 31 91 24 → ~~14 15 18 24 31~~ 39 45 91

---

10 11 12 14 15 17 18 23 24 31

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17 23 33~~ 77 83

39 45 14 18 15 31 91 24 → ~~14 15 18 24 31~~ 39 45 91

---

10 11 12 14 15 17 18 23 24 31 33

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33~~ 77 83

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39~~ 45 91

---

10 11 12 14 15 17 18 23 24 31 33 39

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33~~ 77 83

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45~~ 91

---

10 11 12 14 15 17 18 23 24 31 33 39 45

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 → ~~10 11 12 17 23 33 77~~ 83

39 45 14 18 15 31 91 24 → ~~14 15 18 24 31 39 45~~ 91

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33 77 83~~

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45~~ 91

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77 83

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33 77 83~~

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45 91~~

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77 83 91

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33 77 83~~

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45 91~~

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77 83 91

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33 77 83~~

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45 91~~

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77 83 91

$C(1) = 1$

$C(n) = 2\ C(n/2) + n$

$C(n) = n \log_2(n)$

Trier par dichotomie-fusion

12 17 10 23 33 77 83 11 39 45 14 18 15 31 91 24

---

12 17 10 23 33 77 83 11 ⟶ ~~10 11 12 17 23 33 77 83~~

39 45 14 18 15 31 91 24 ⟶ ~~14 15 18 24 31 39 45 91~~

---

10 11 12 14 15 17 18 23 24 31 33 39 45 77 83 91

$C(1) = 1$

$C(n) = 2\,C(n/2) + n$

$C(n) = n \log_2(n)$

$C(1\,000) = 10\,000$

$C(1\,000\,000) = 20\,000\,000$

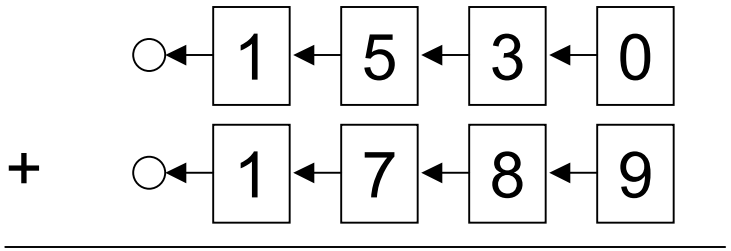
Les 4 opérations

parallèle synchrone : opérateur circuit
séquentiel : passage en précision arbitraire
parallèle asynchrone : hors sujet

- Addition / soustraction
  - circuit: temps log(n), surface / énergie n log(n)
  - séquentiel : temps / énergie n, mémoire n
- Multiplication
  - circuit: temps log(n), surface / énergie n^2
    (en utilisant des bases redondantes)
  - séquentiel: plus compliqué (cf J. Vuillemin)
- Division
  - hors de portée de ce cours...

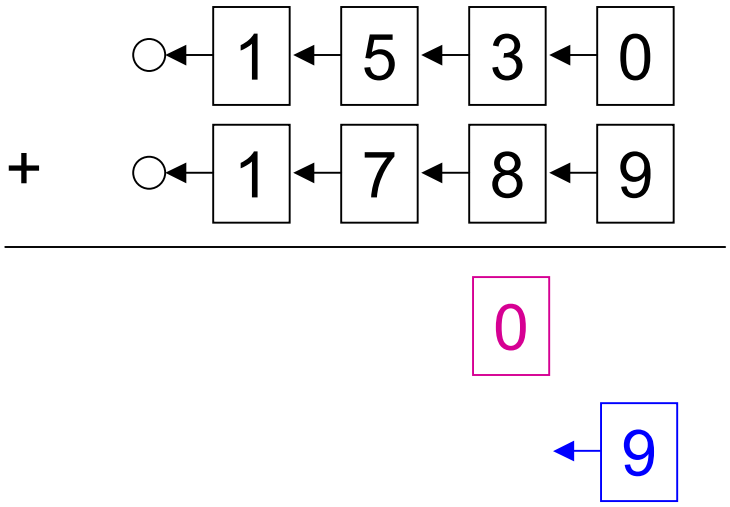
Séquentiel : listes chaînées

1 530 + 1 789 = 3 319

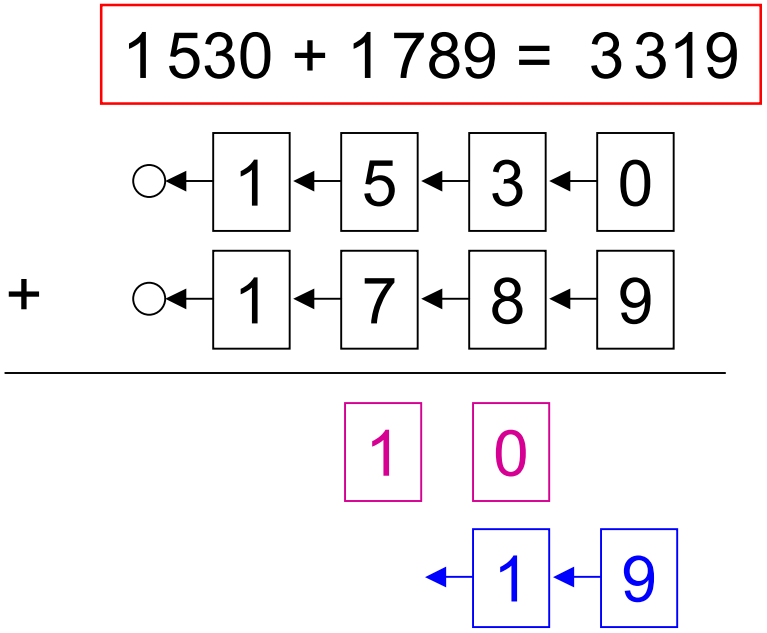


Séquentiel : listes chaînées

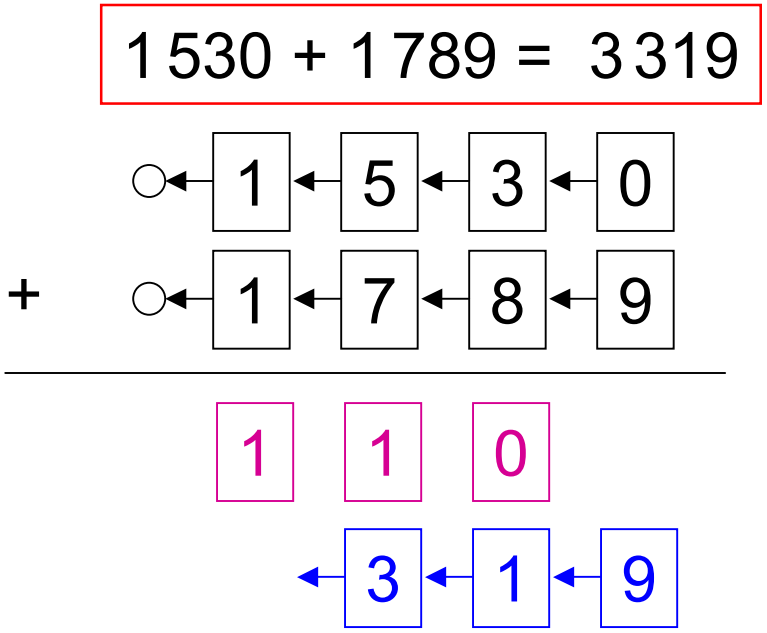
1 530 + 1 789 = 3 319



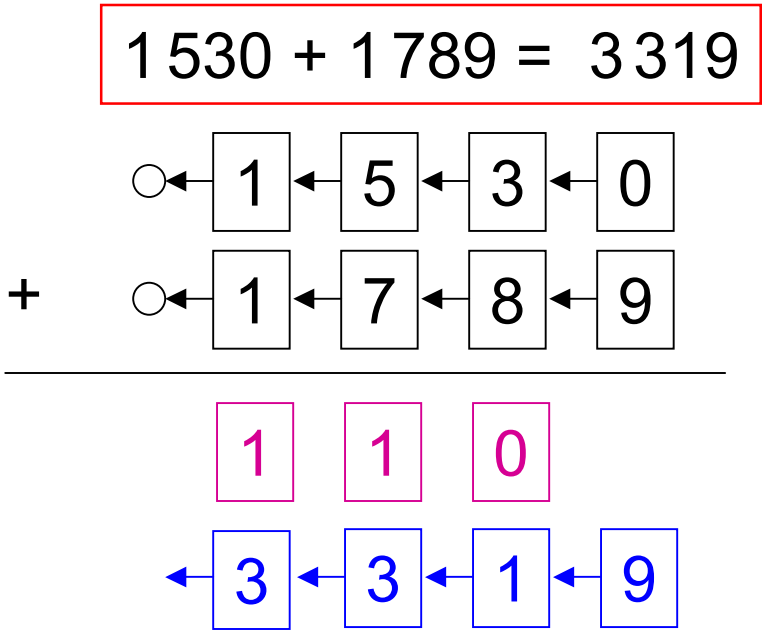
Séquentiel : listes chaînées



Séquentiel : listes chaînées

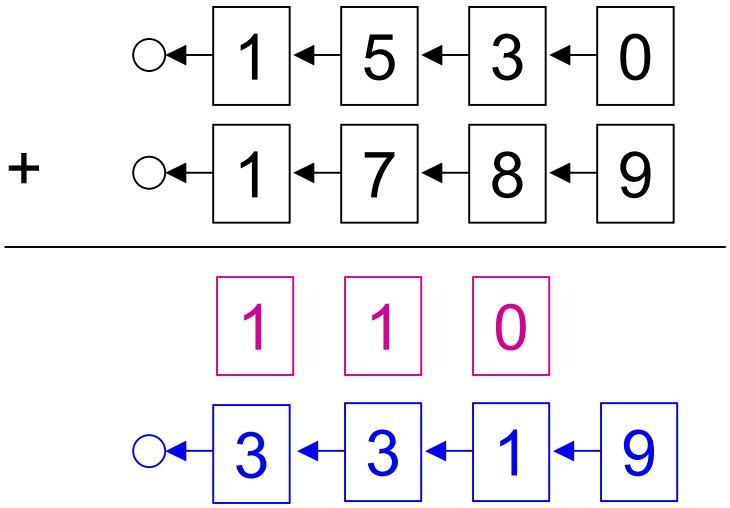


Séquentiel : listes chaînées



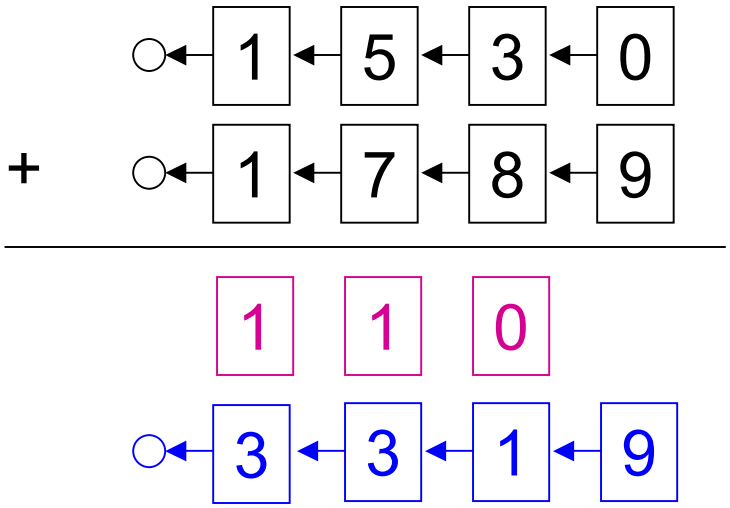
Séquentiel : listes chaînées

1 530 + 1 789 = 3 319

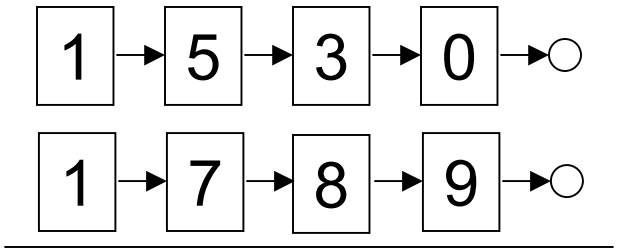


Séquentiel : listes chaînées

1 530 + 1 789 = 3 319

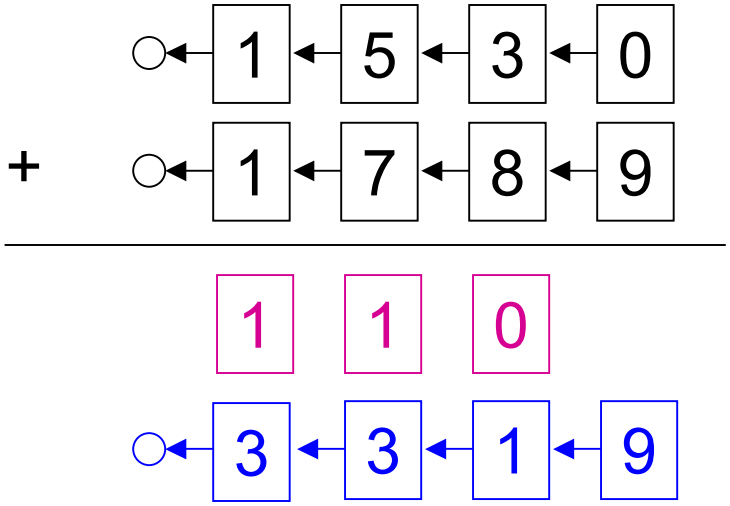


1 530 < 1 789 ?

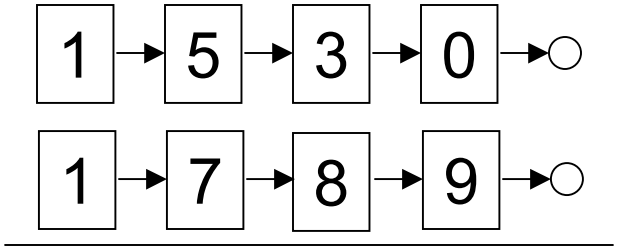


Séquentiel : listes chaînées

1 530 + 1 789 = 3 319



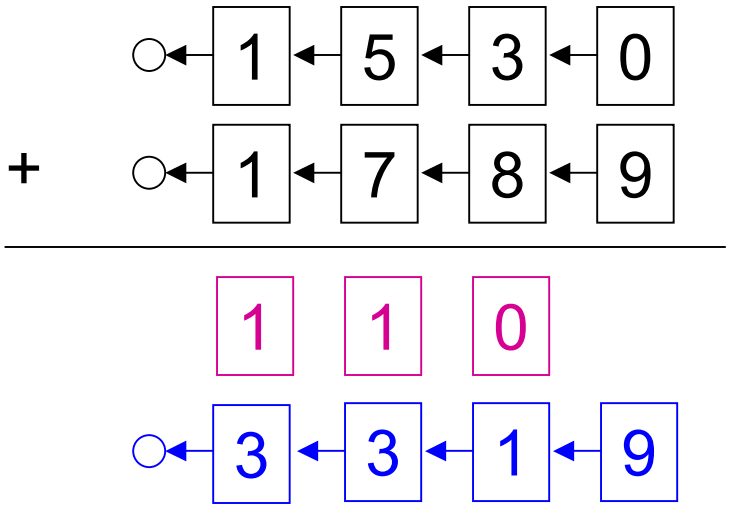
1 530 < 1 789 ?



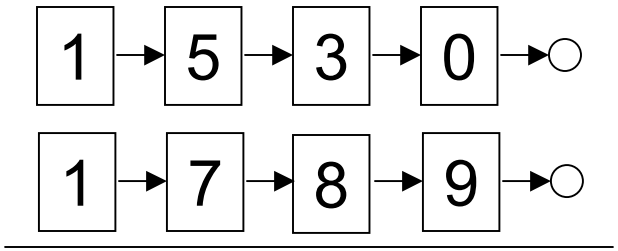
F

Séquentiel : listes chaînées

1 530 + 1 789 = 3 319



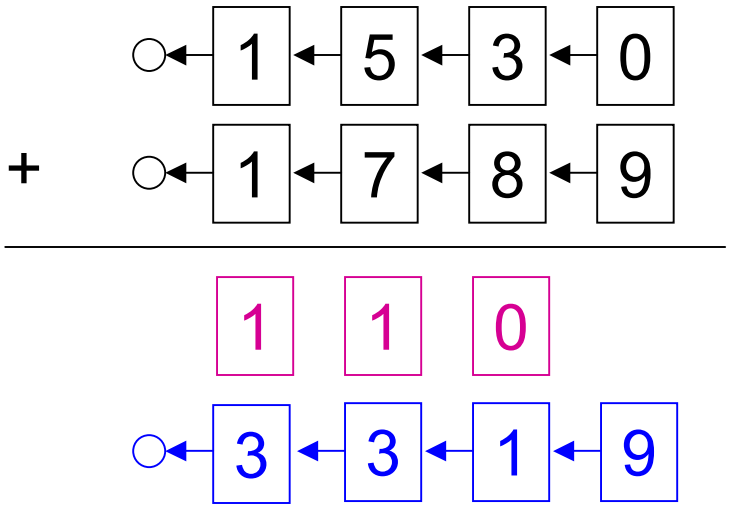
1 530 < 1 789 ?



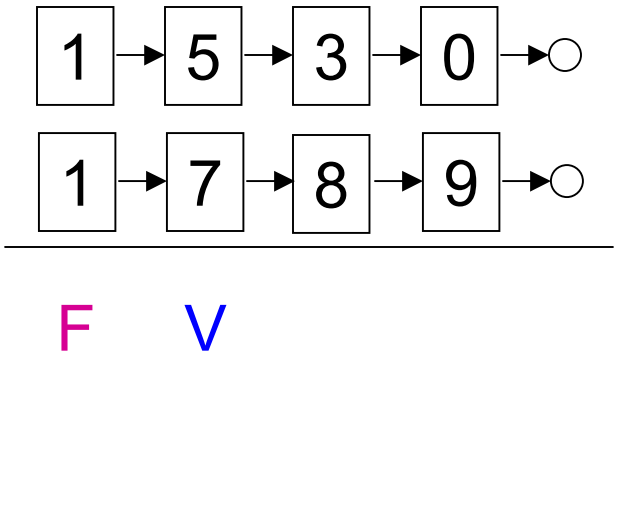
F V

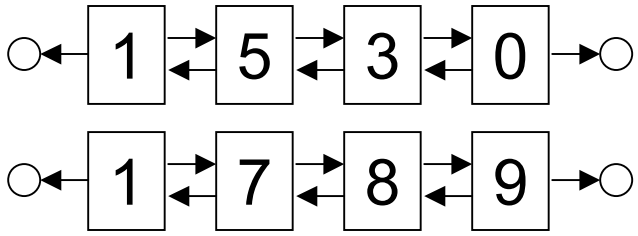
Séquentiel : listes chaînées

1 530 + 1 789 = 3 319



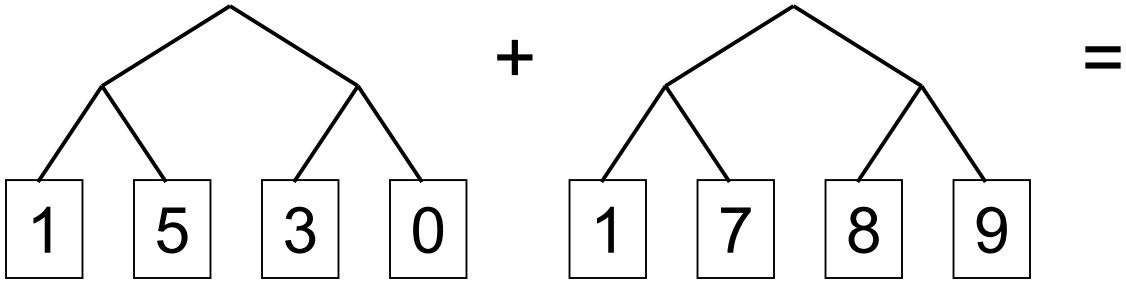
1 530 < 1 789 ?



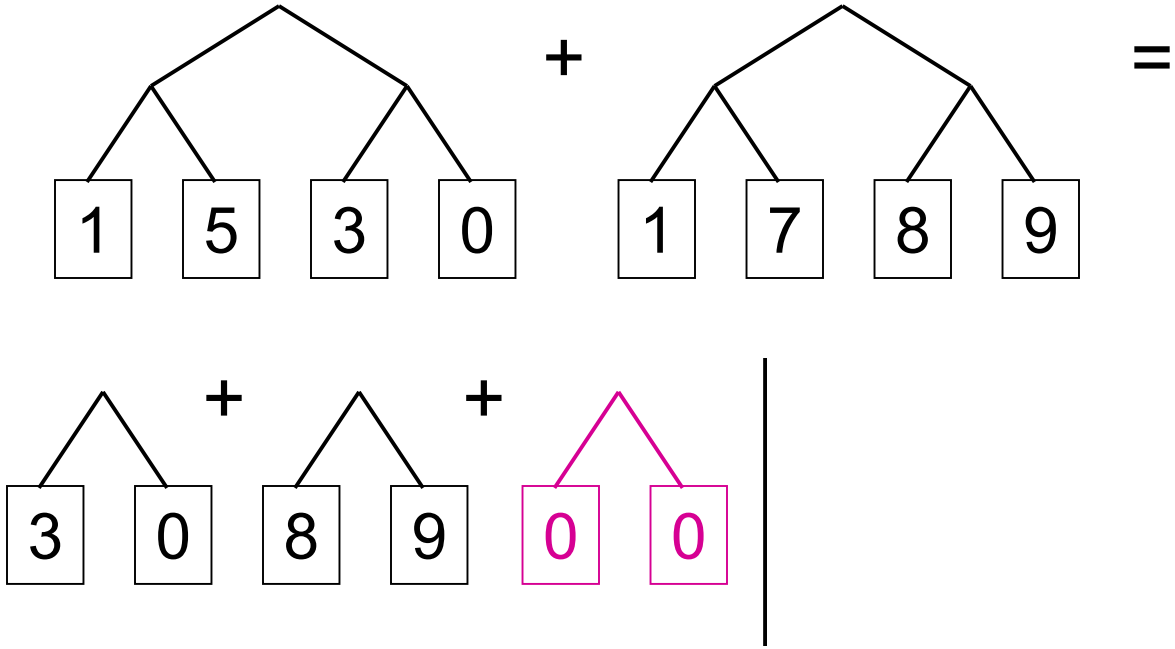


listes doublement chaînées

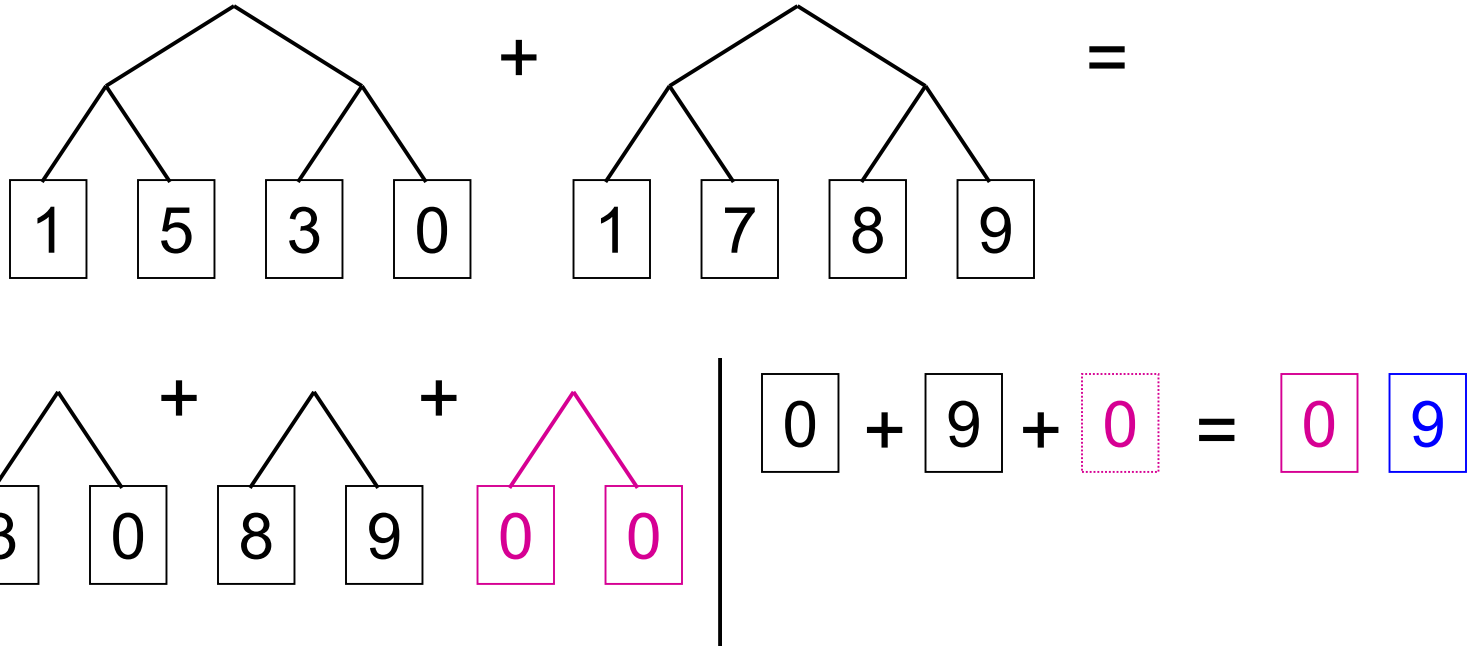
Arbres binaires



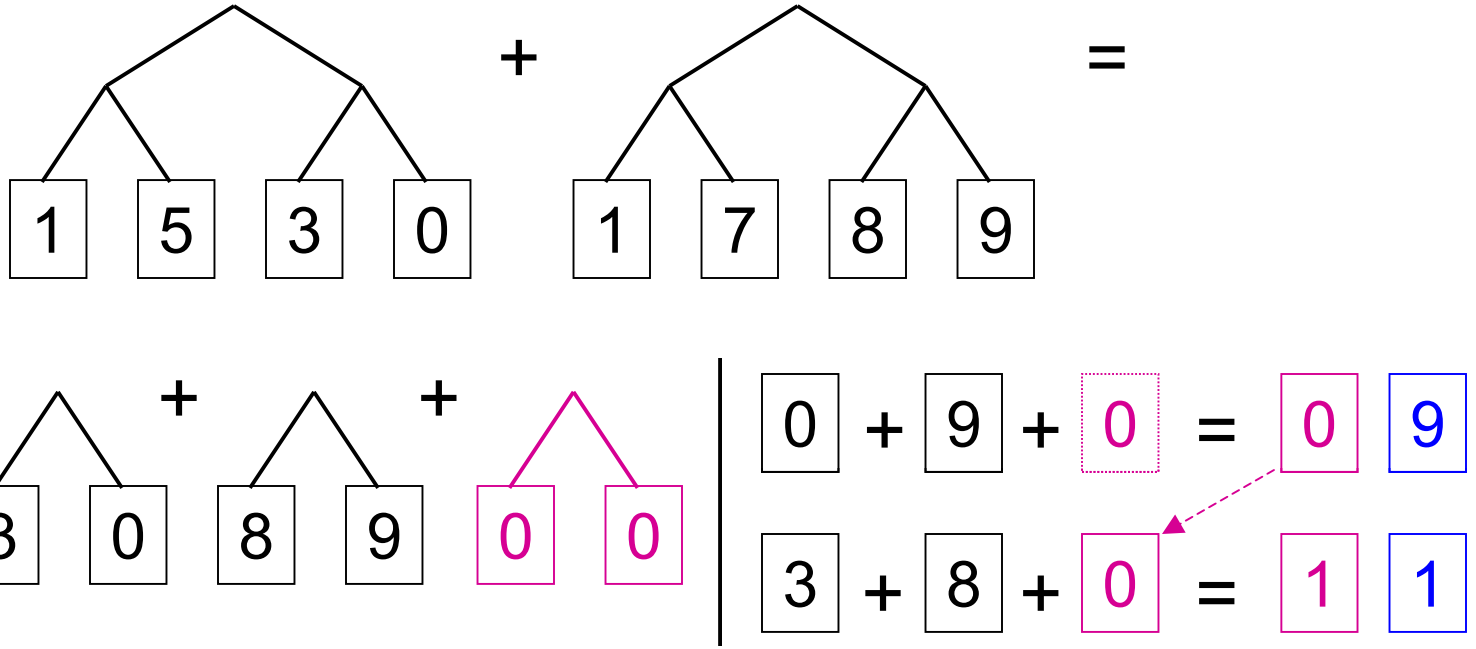
Arbres binaires



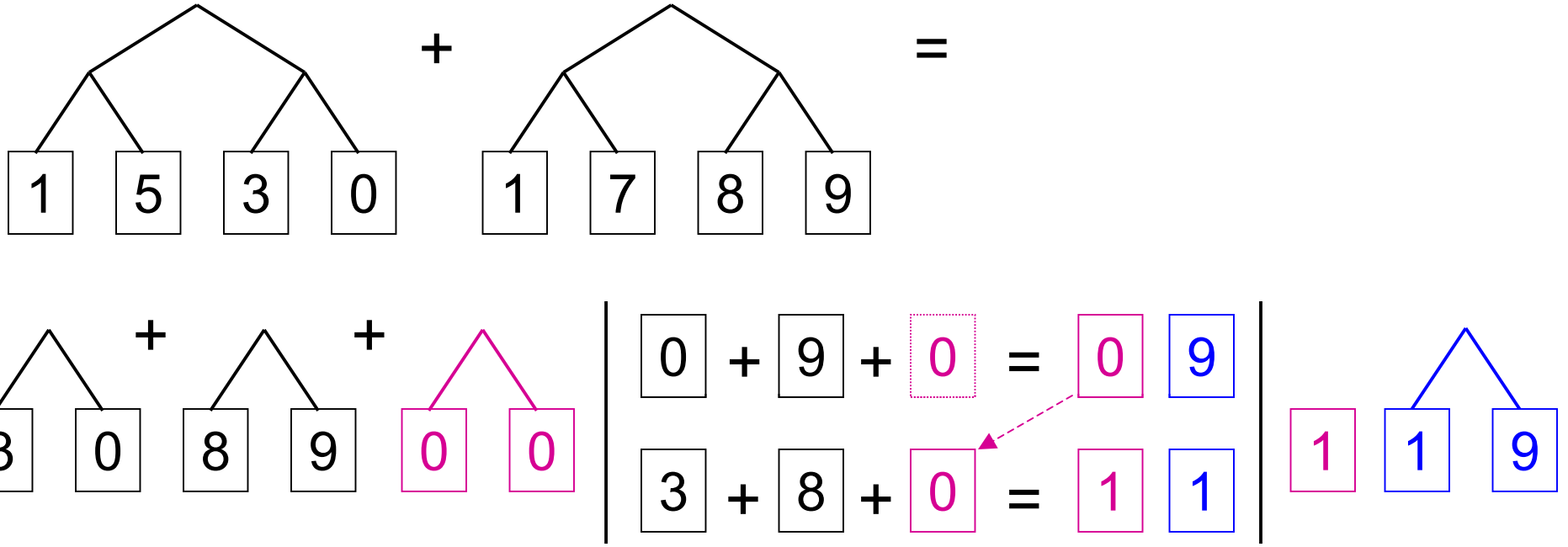
Arbres binaires



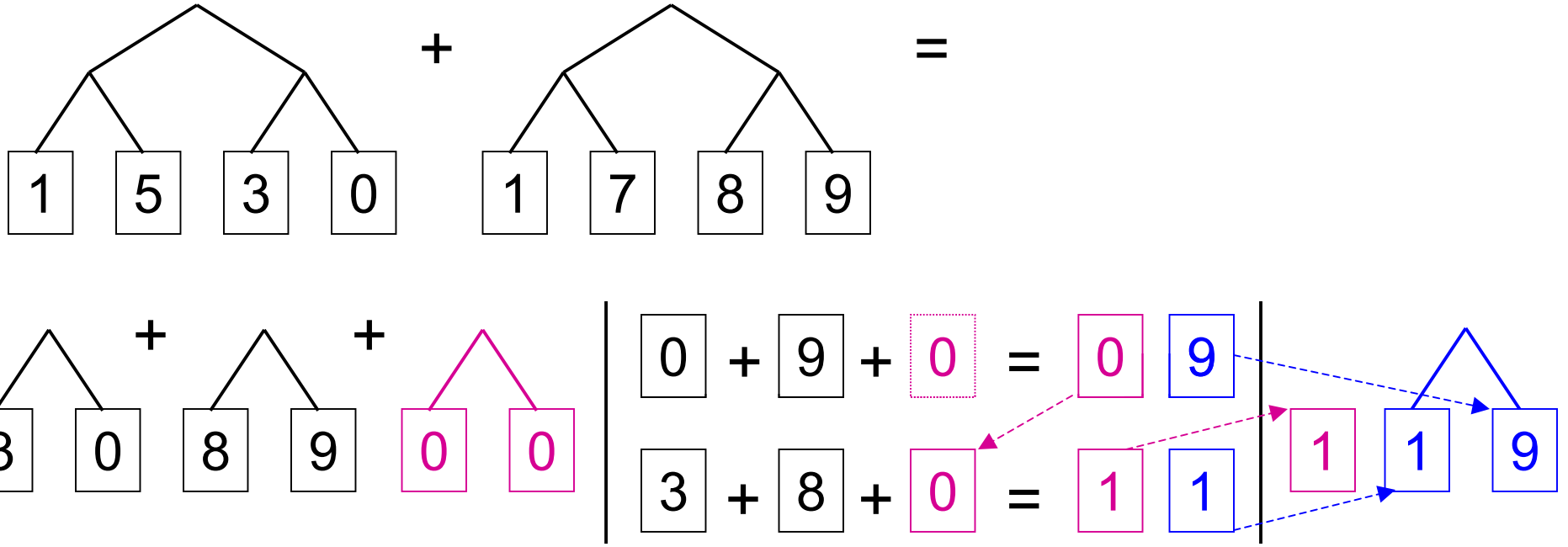
Arbres binaires



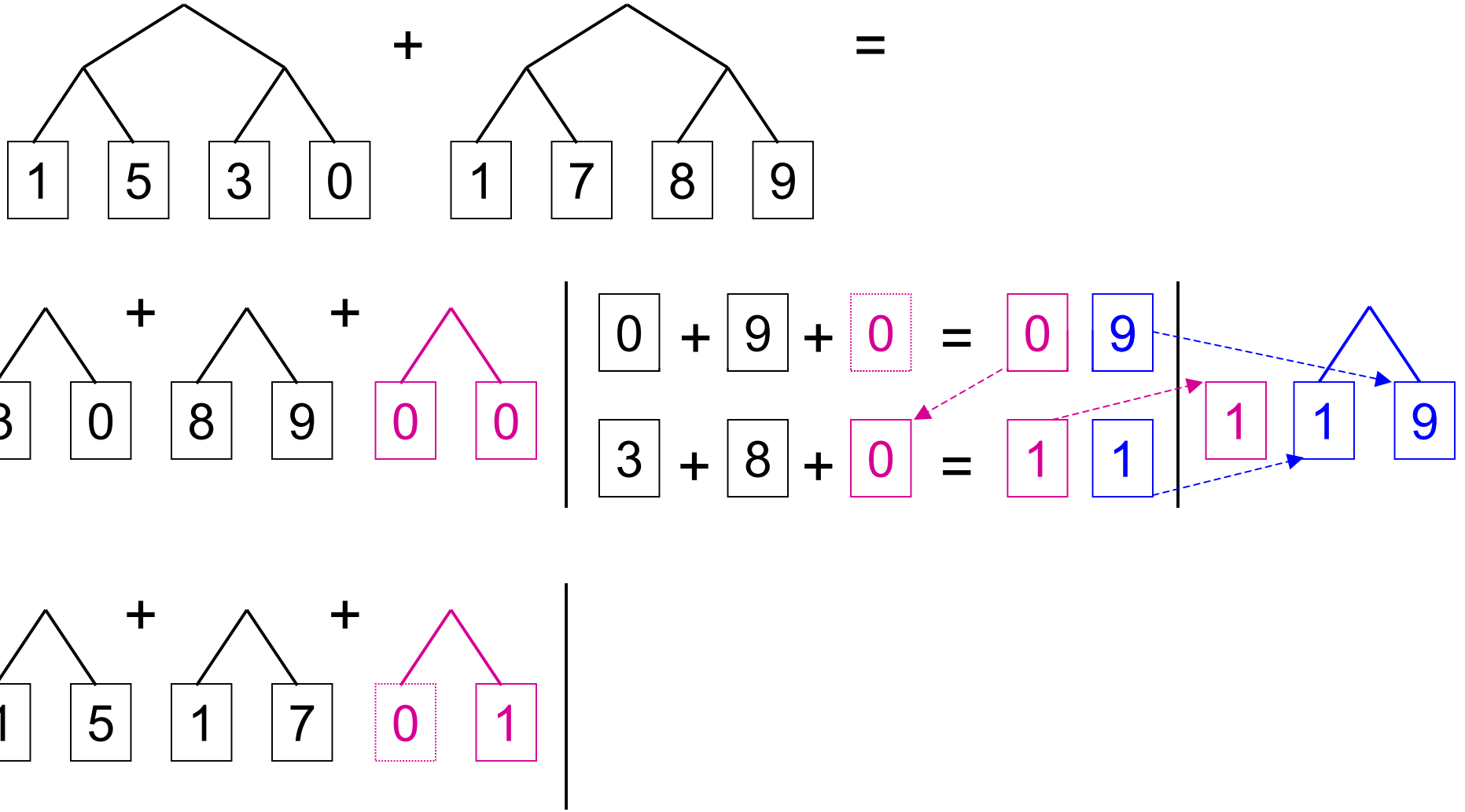
Arbres binaires



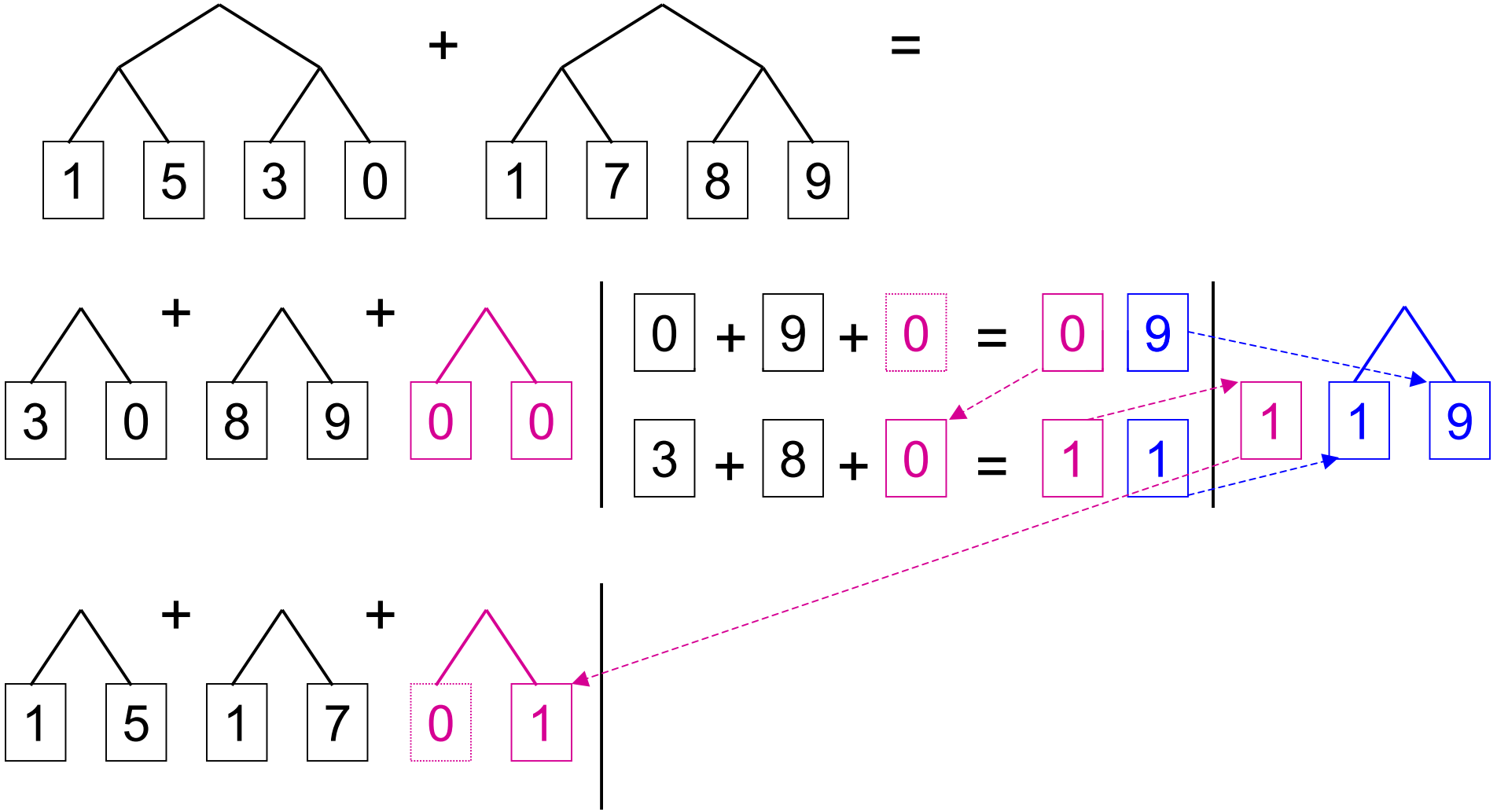
Arbres binaires



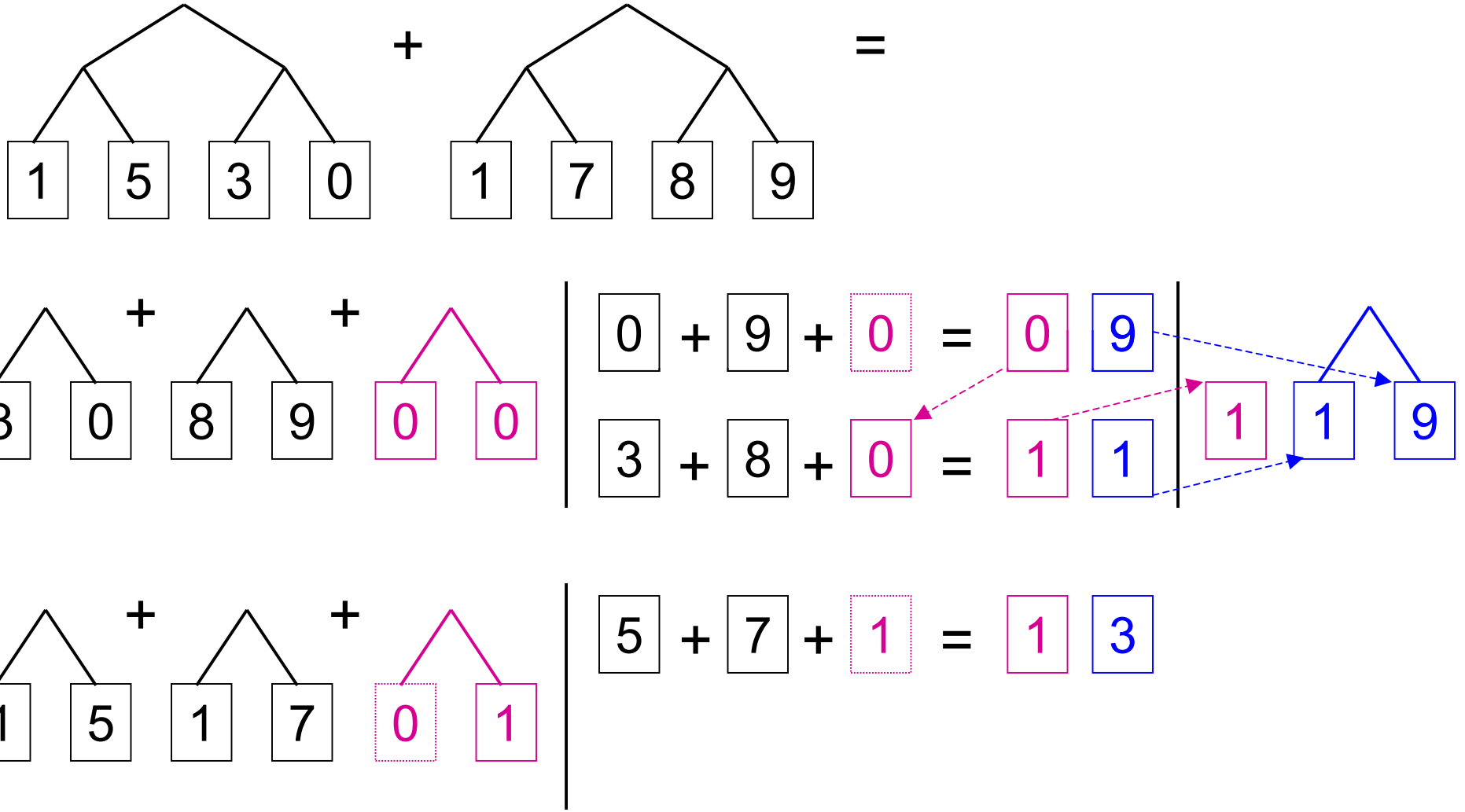
Arbres binaires



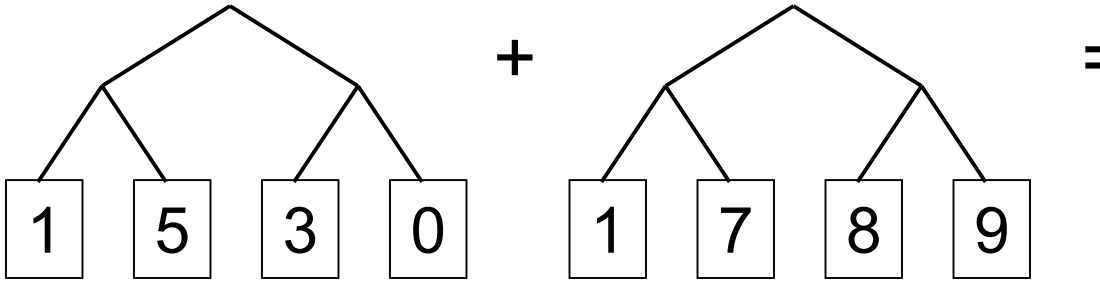
Arbres binaires

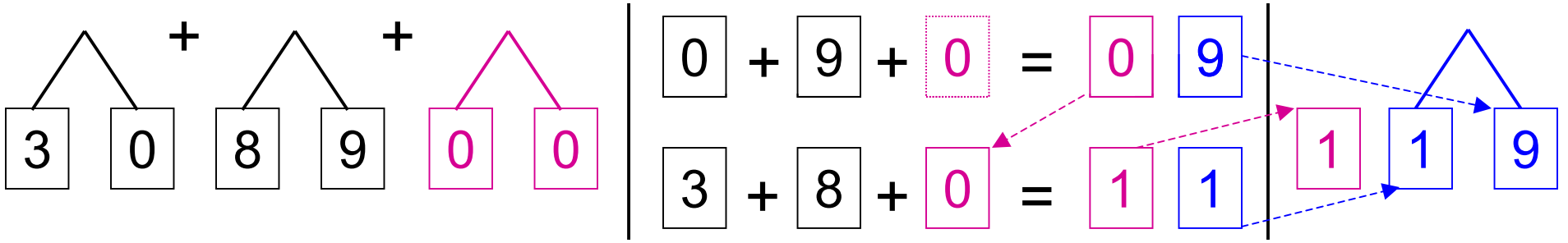


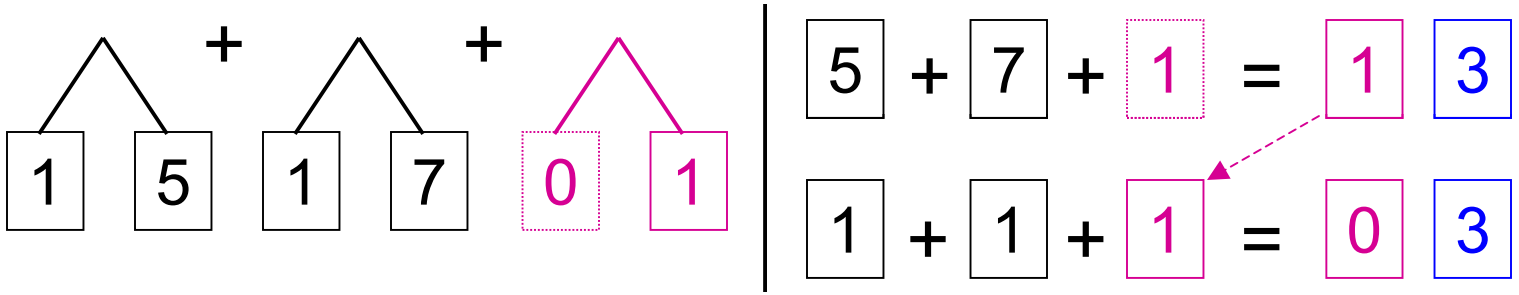
Arbres binaires



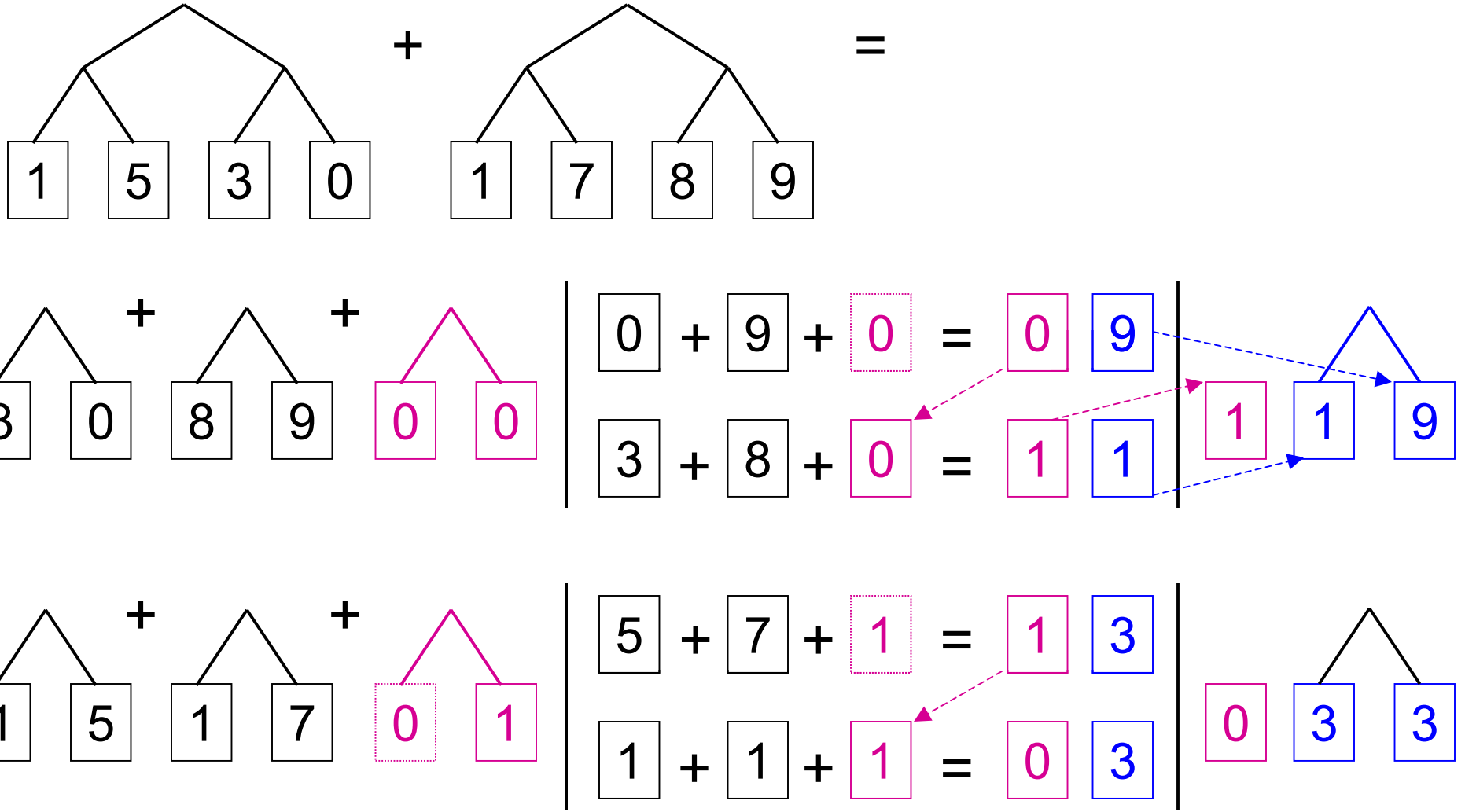
Arbres binaires



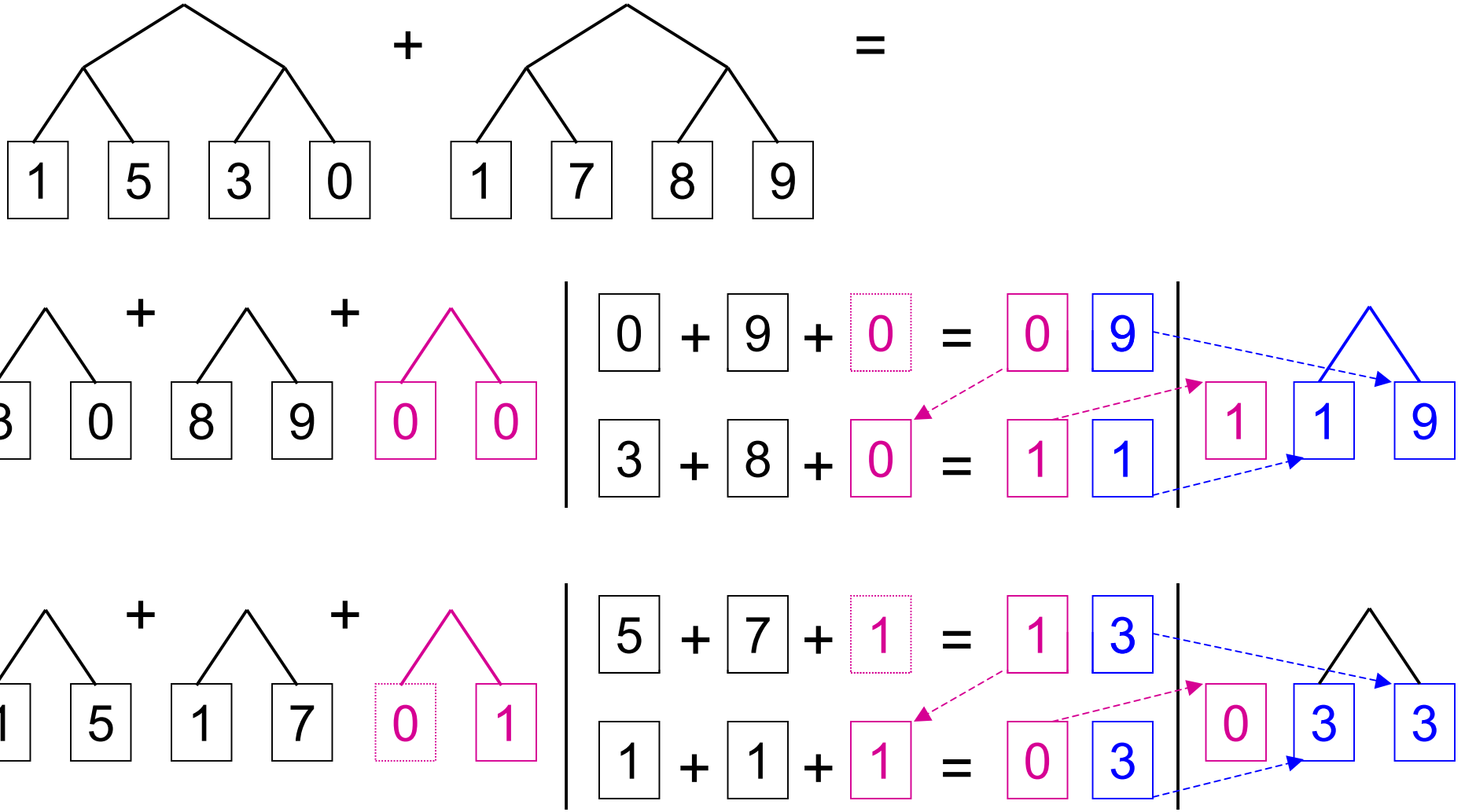




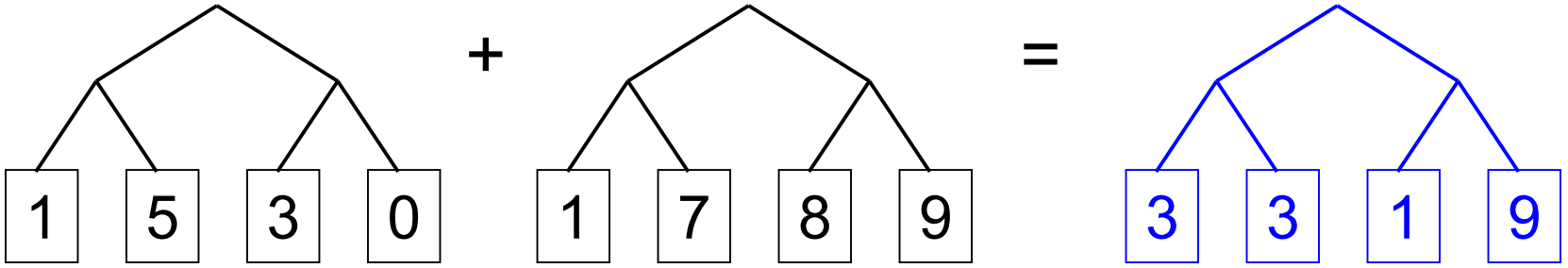
Arbres binaires

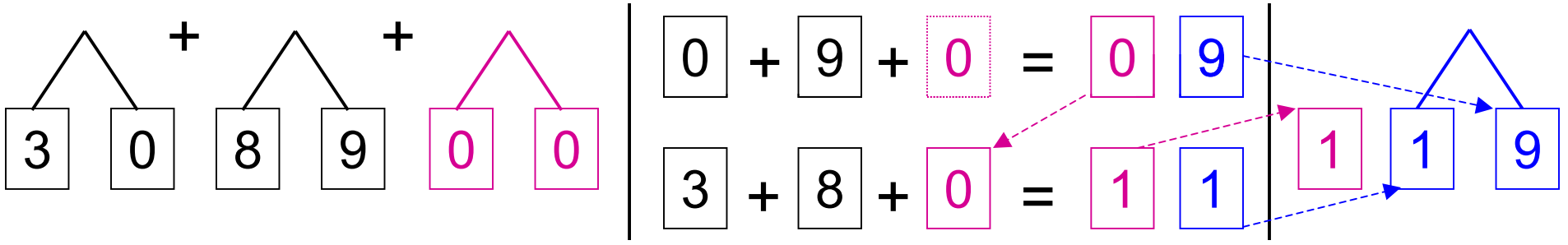


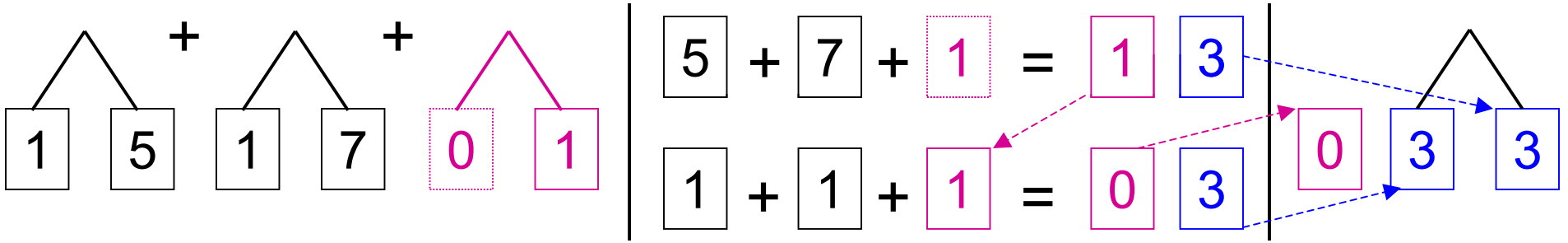
Arbres binaires



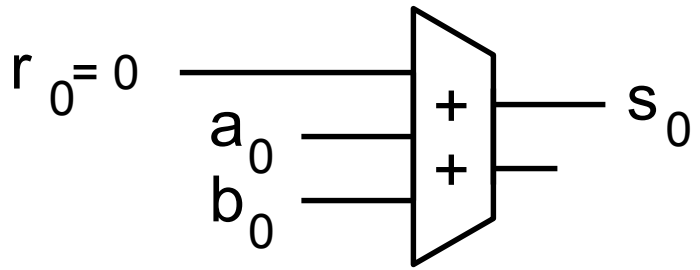
Arbres binaires



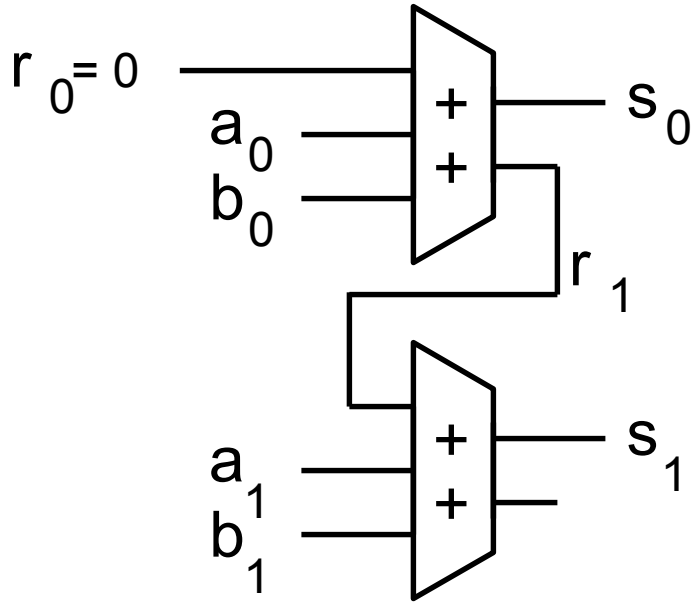




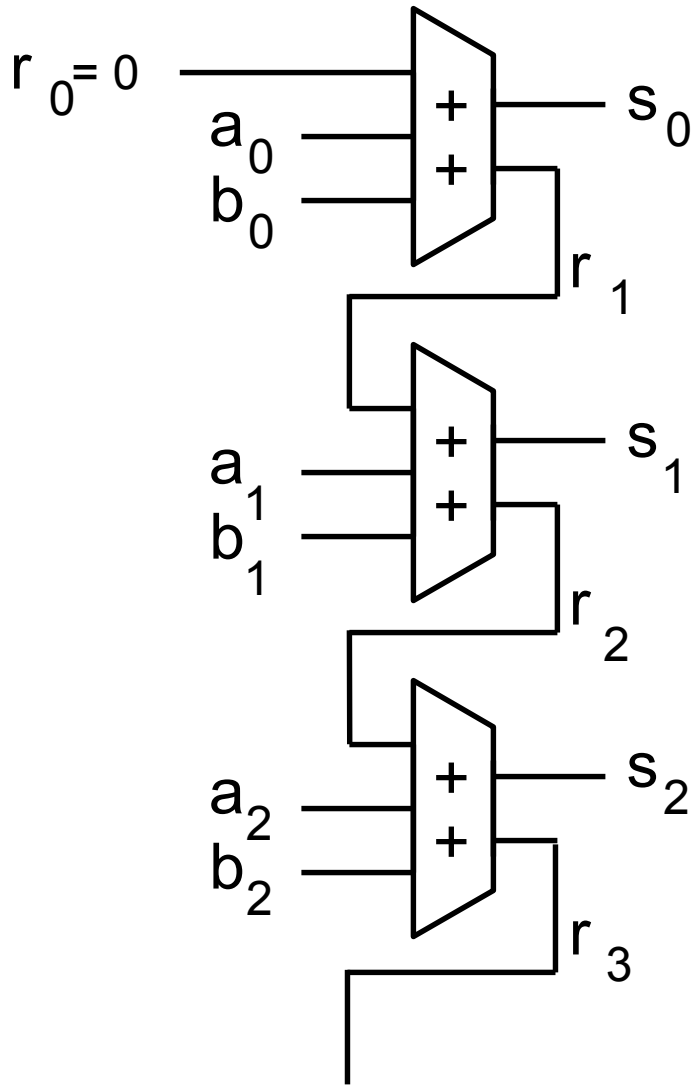
Circuit : propagation de retenue



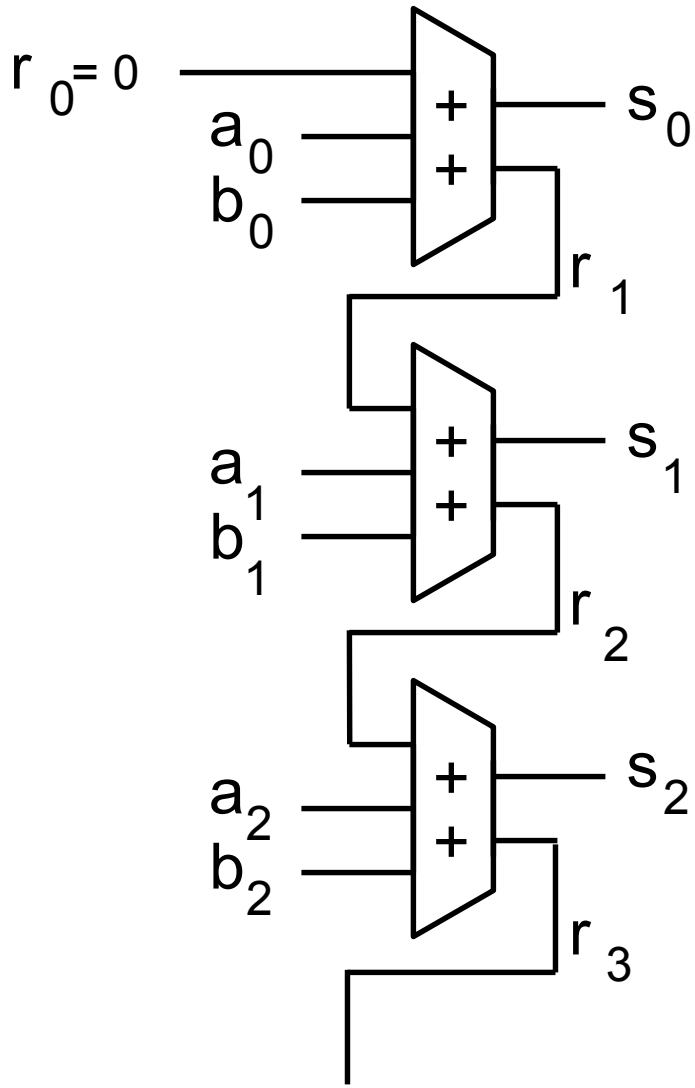
Circuit : propagation de retenue



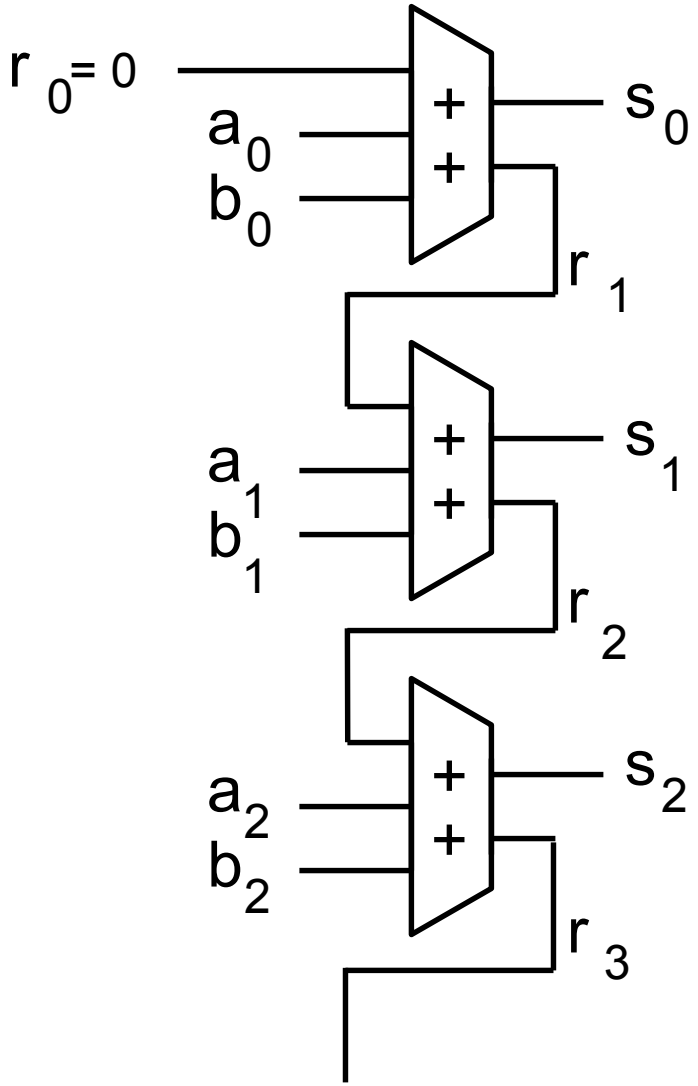
Circuit : propagation de retenue



Circuit : propagation de retenue



Circuit : propagation de retenue



Pour n bits :

espace n

temps n

énergie n

Addition logarithmique (von Neumann)

- Calculer en même temps a+b et a+b+1
- Propager par structure dichotomique

s = a+b

s' = a+b+1

1530 + 1789

1530 + 1789

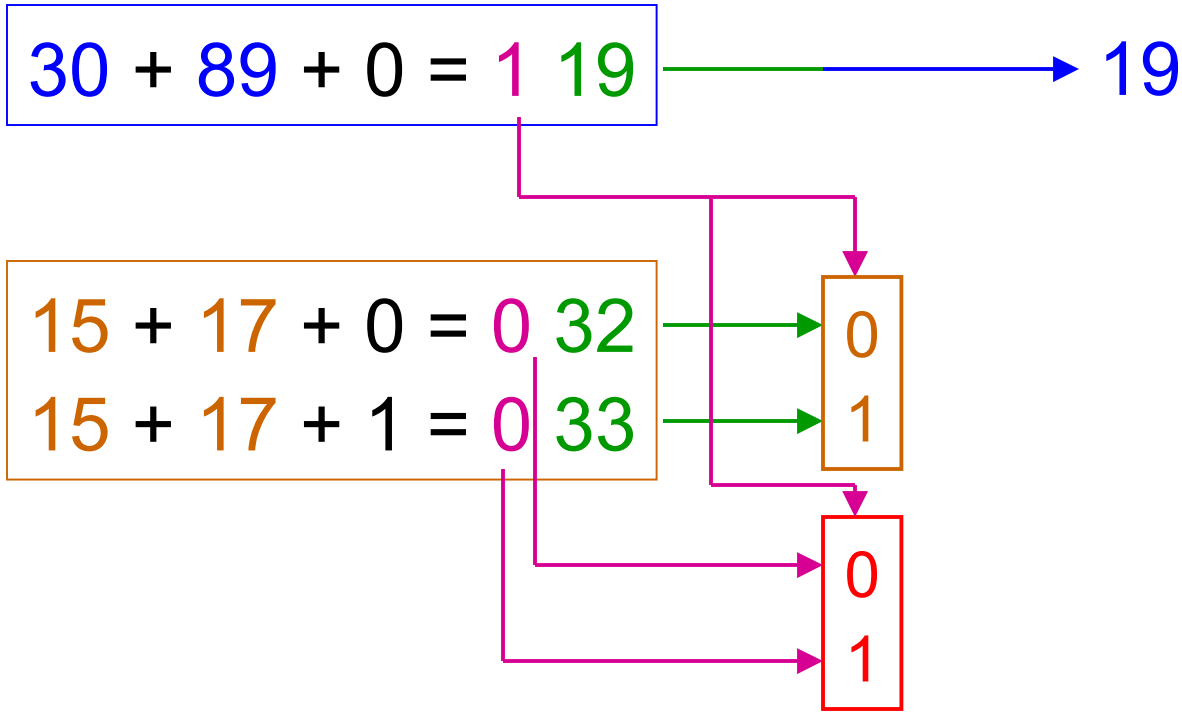
1530 + 1789

30 + 89 + 0 = 1 19

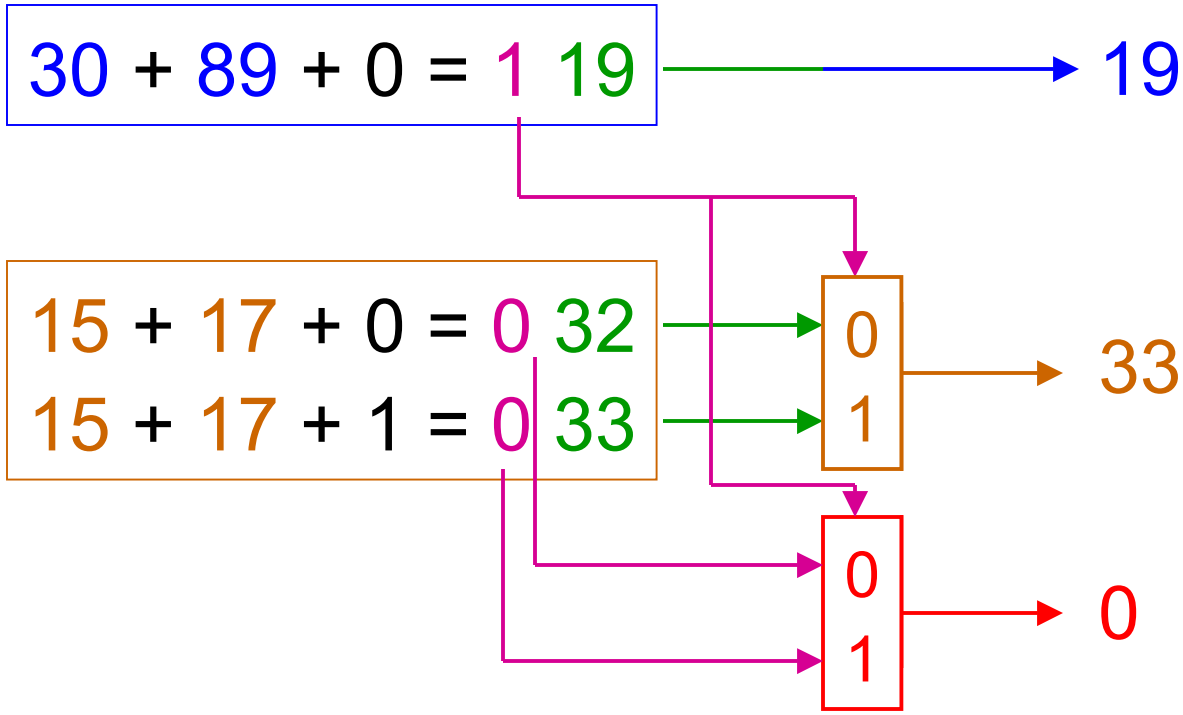
15 + 17 + 0 = 0 32

15 + 17 + 1 = 0 33

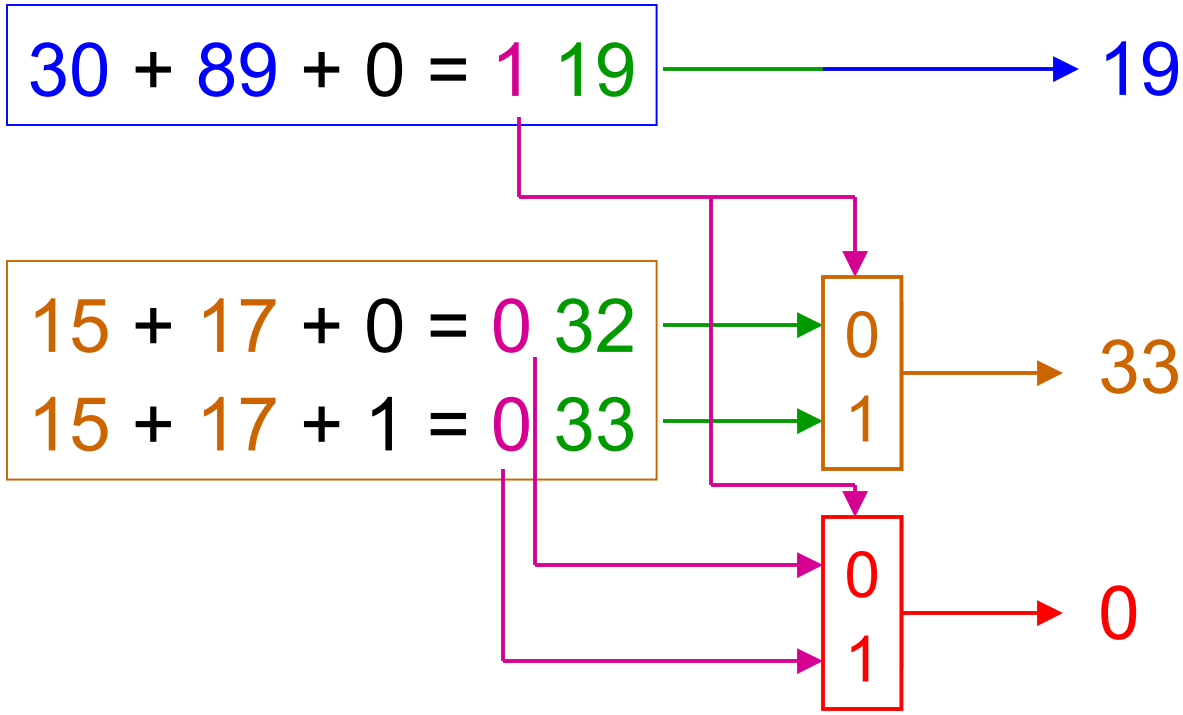
1530 + 1789



1530 + 1789



1530 + 1789 = 03319



30 + 89 + 0

30 + 89 + 0

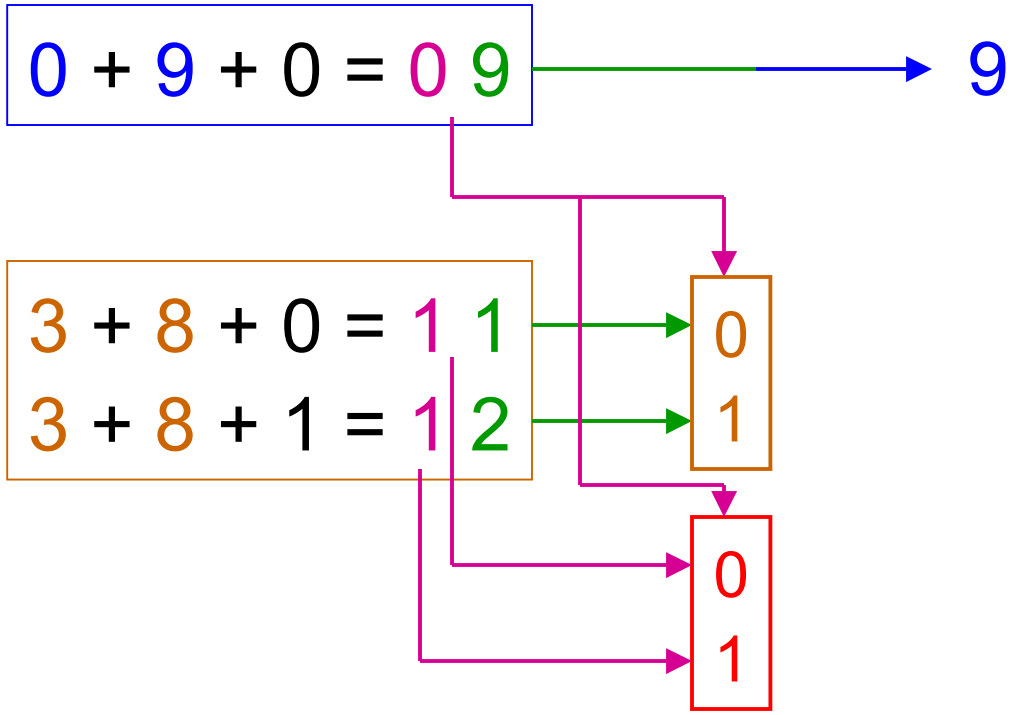
30 + 89 + 0

0 + 9 + 0 = 0 9

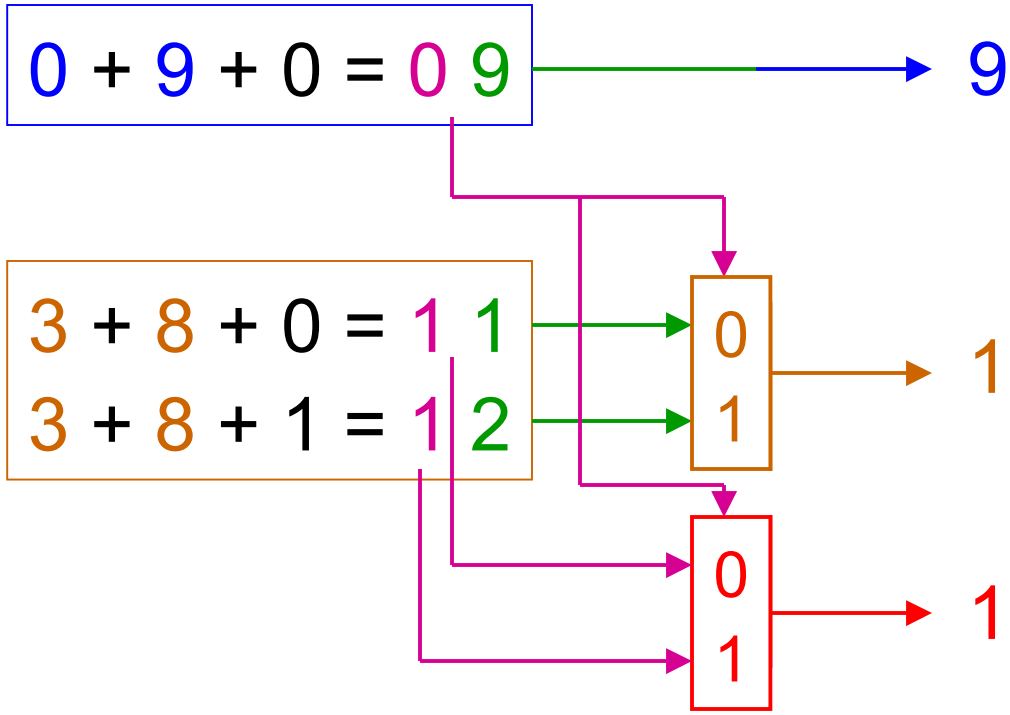
3 + 8 + 0 = 1 1

3 + 8 + 1 = 1 2

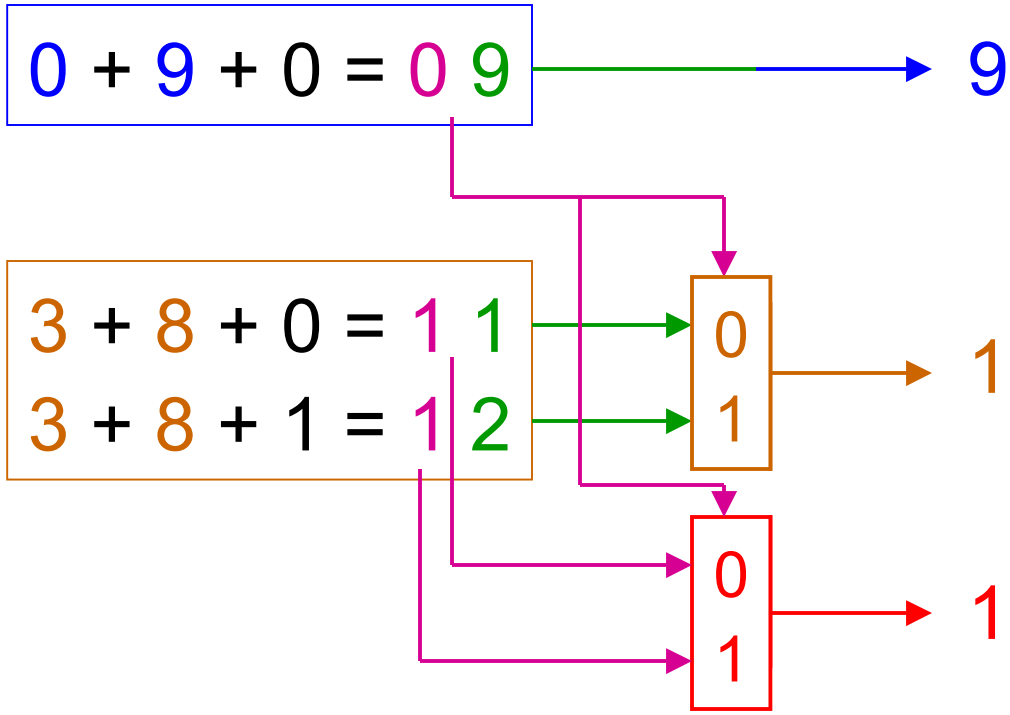
30 + 89 + 0



30 + 89 + 0



30 + 89 + 0 = 119



15 + 17 + 0

15 + 17 + 1

15 + 17 + 0

15 + 17 + 1

5 + 7 + 0 = 1 2

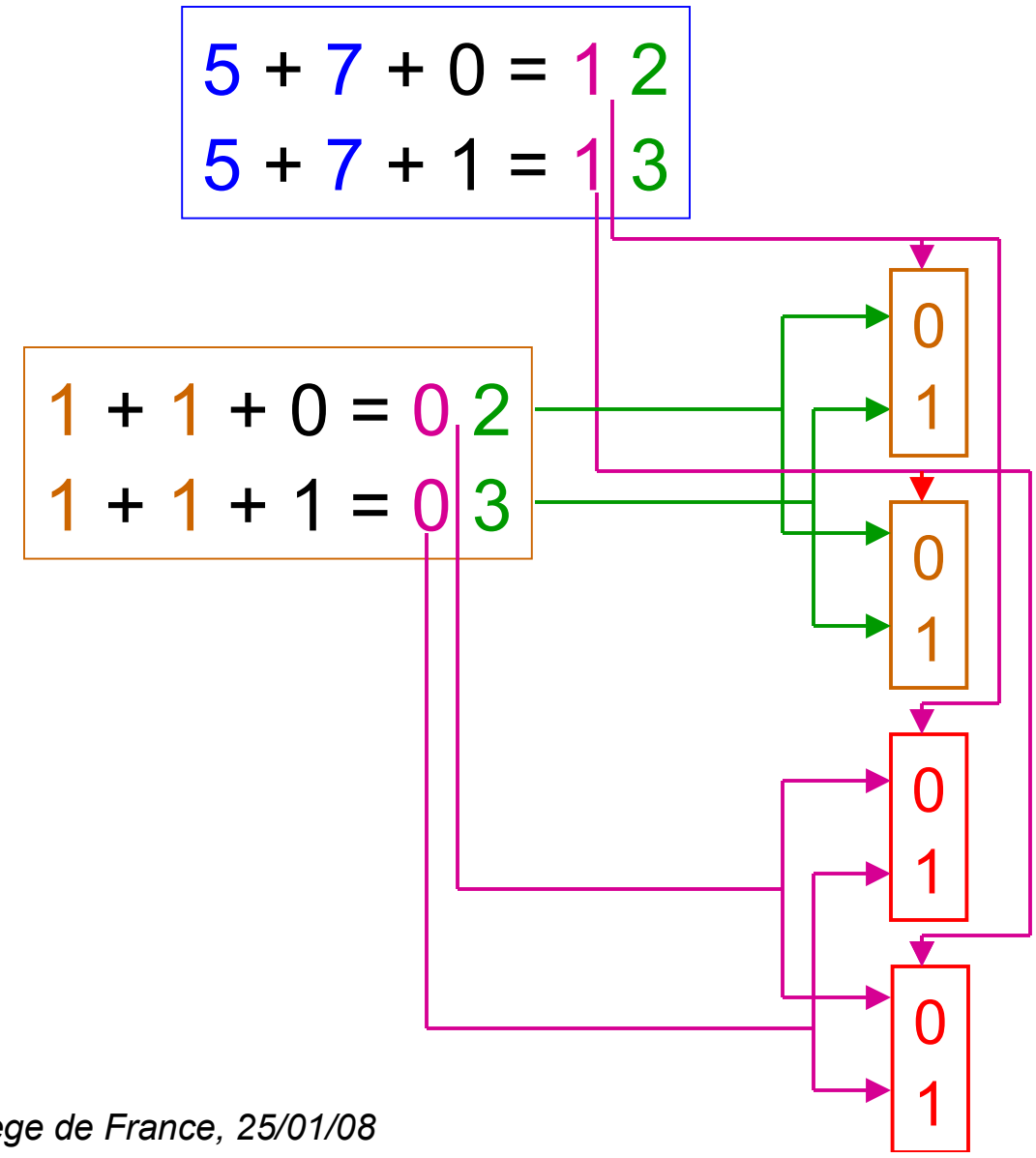
5 + 7 + 1 = 1 3

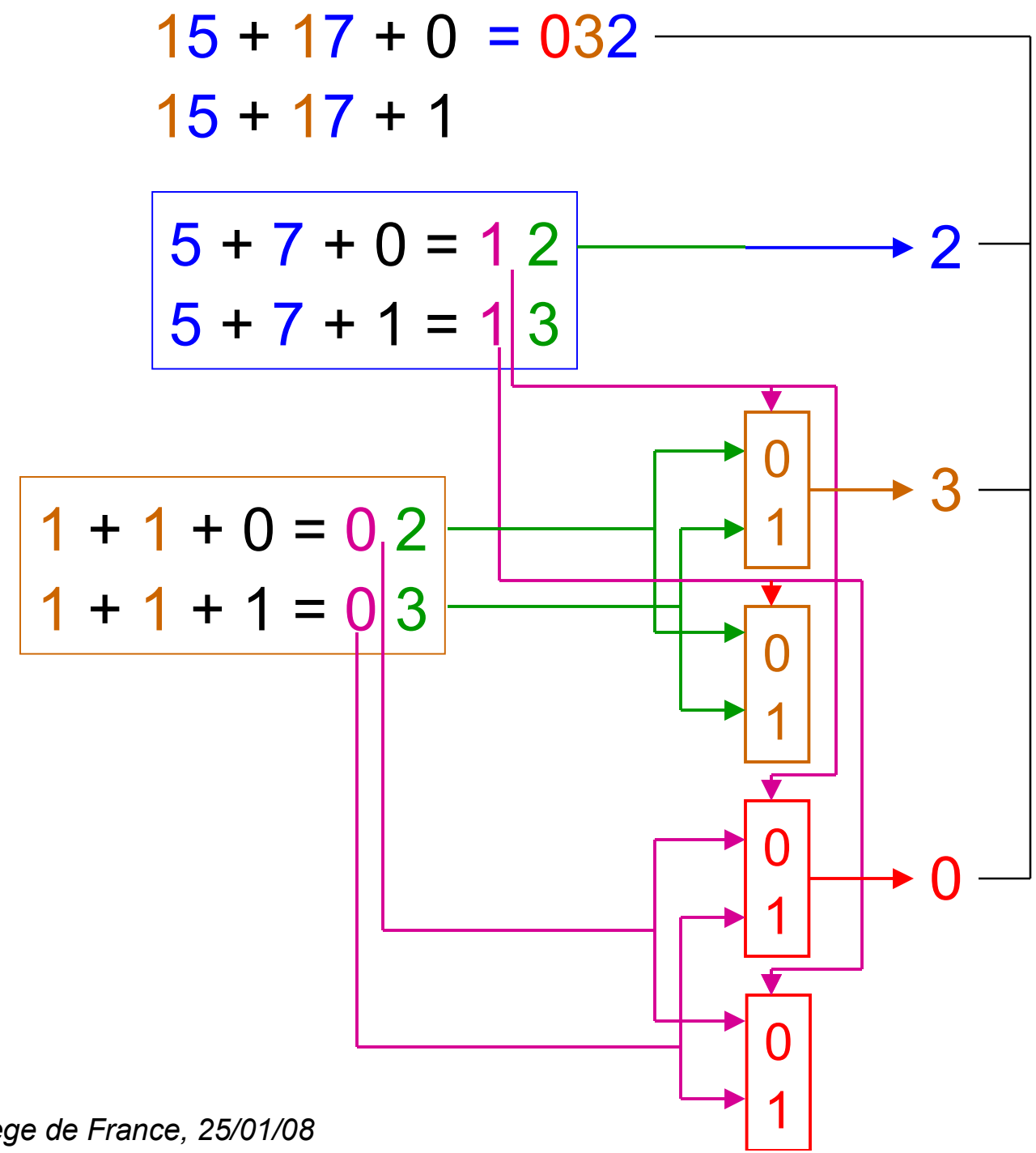
1 + 1 + 0 = 0 2

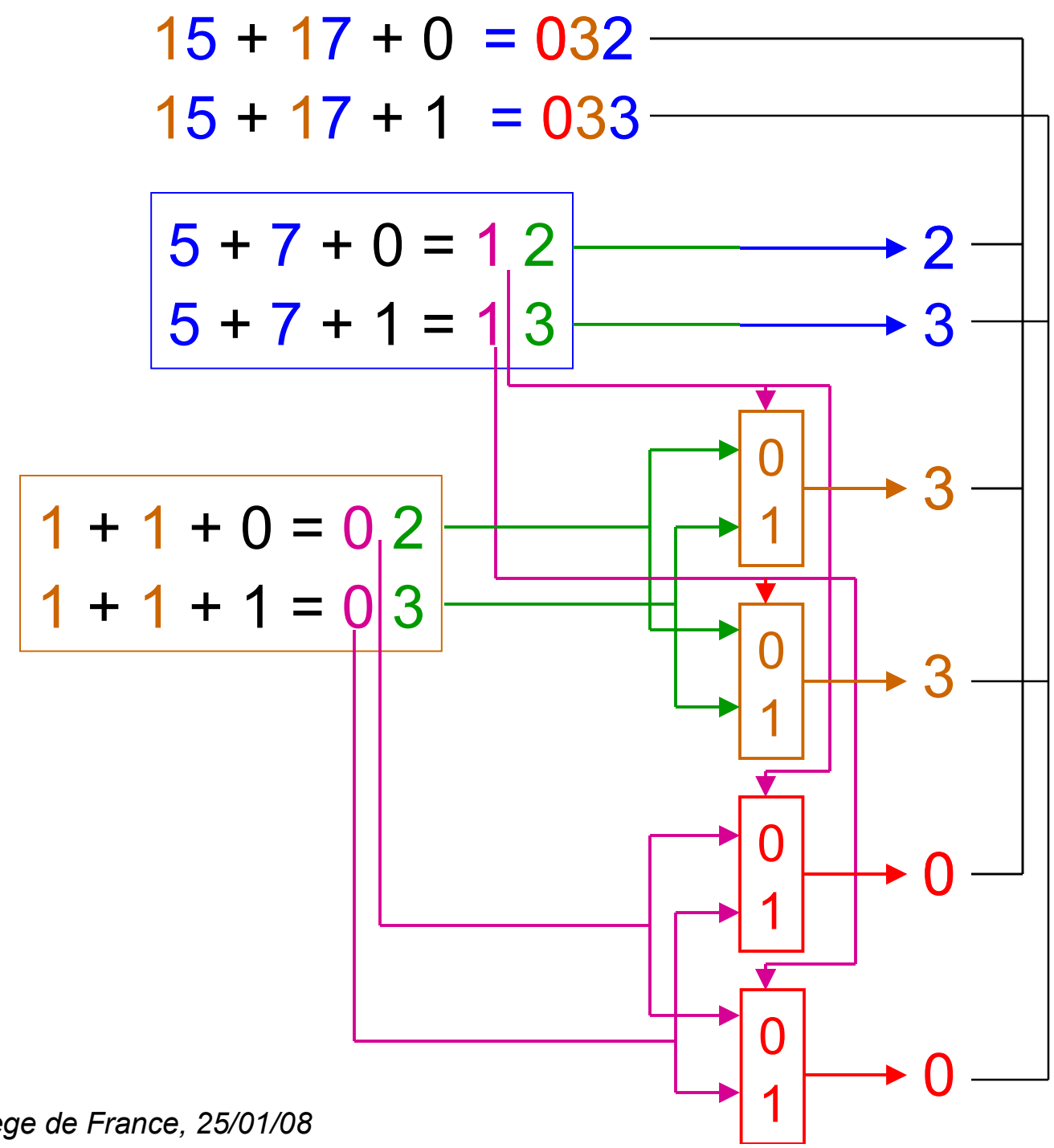
1 + 1 + 1 = 0 3

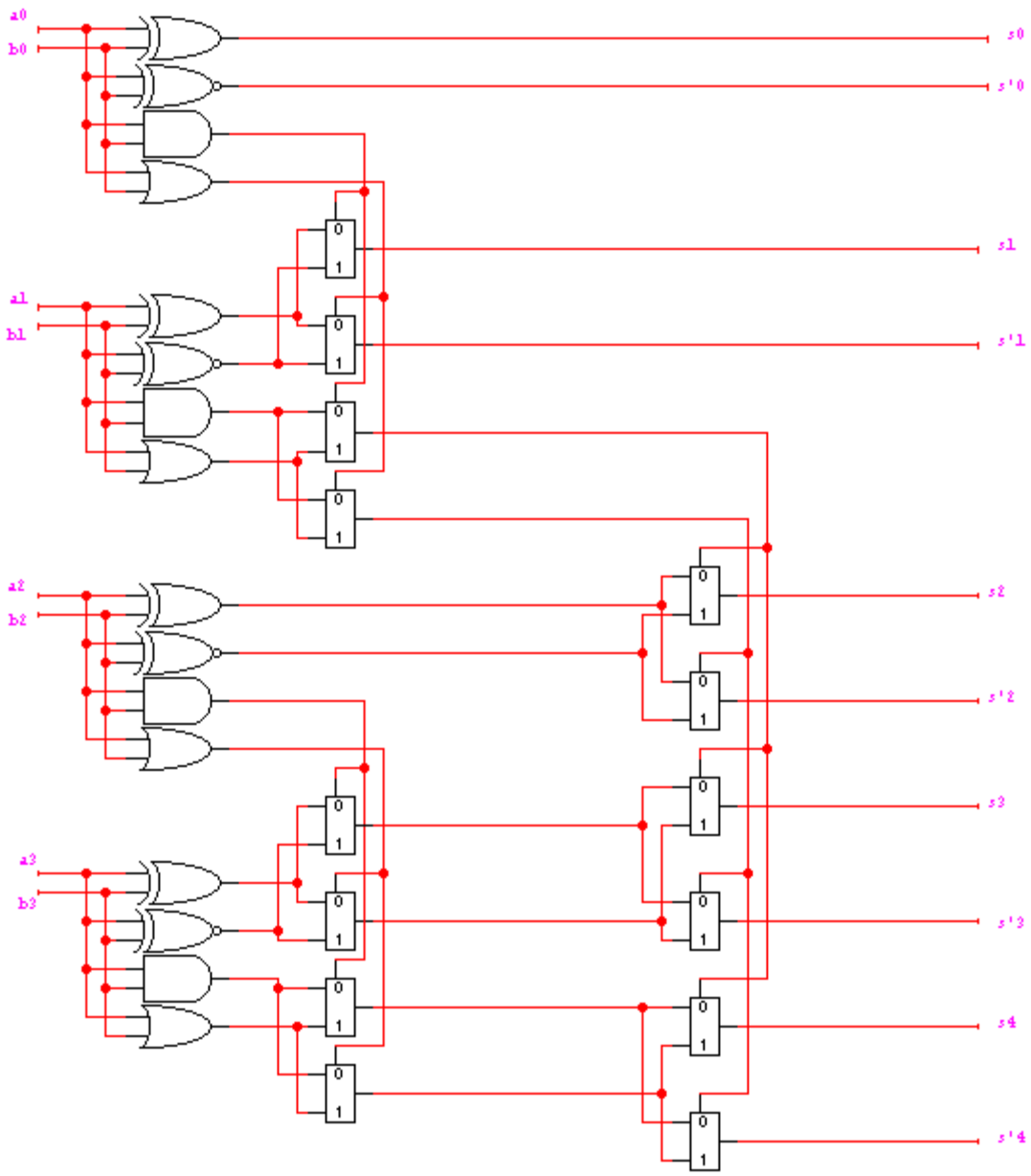
15 + 17 + 0

15 + 17 + 1

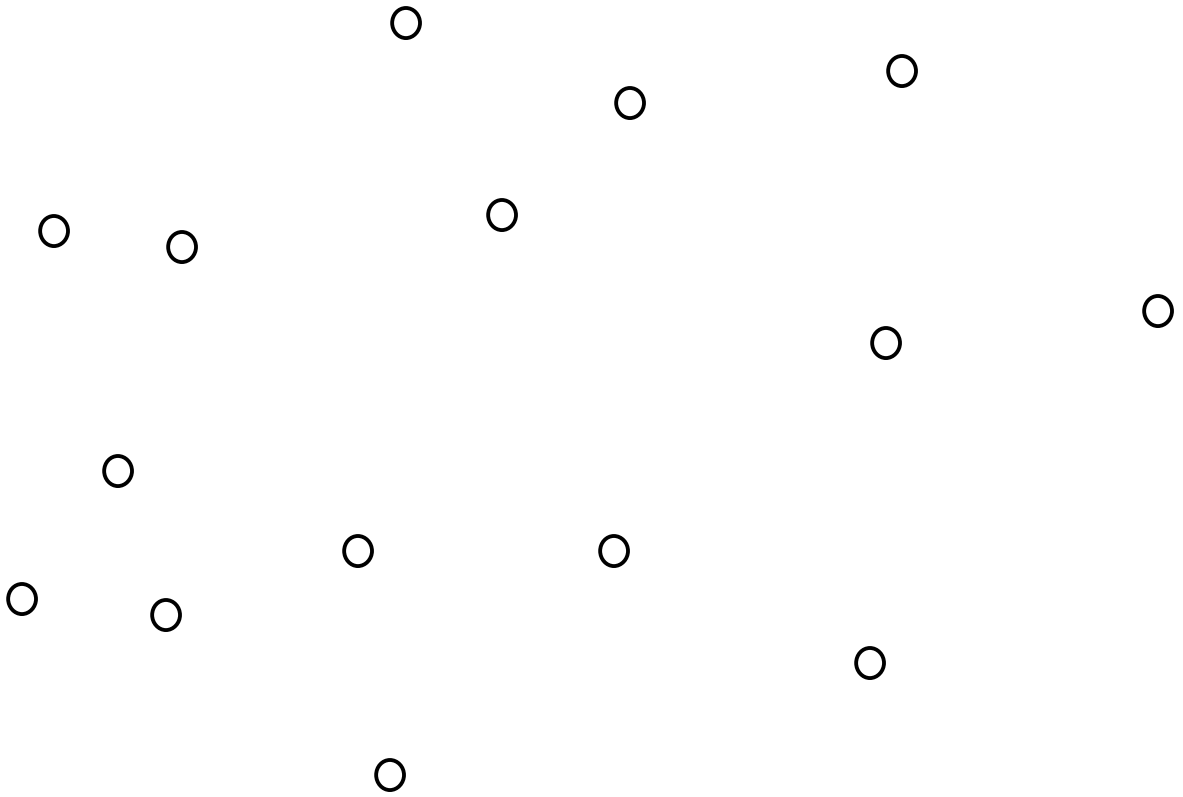



15 + 17 + 0 = 032
15 + 17 + 1
5 + 7 + 0 = 1 2
5 + 7 + 1 = 1 3
2
1 + 1 + 0 = 0 2
1 + 1 + 1 = 0 3
0
1
3
0
1
0
1
0
1

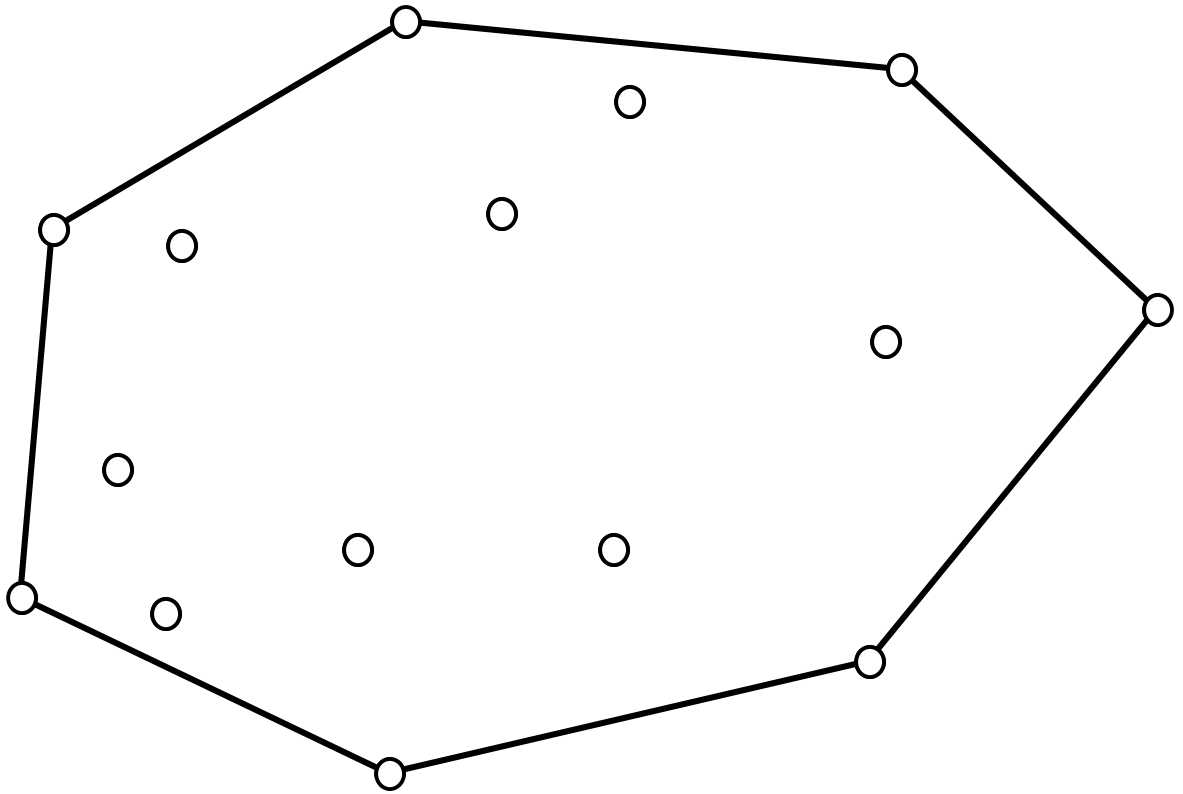

15 + 17 + 0 = 032
15 + 17 + 1 = 033
5 + 7 + 0 = 1 2
5 + 7 + 1 = 1 3
2
3
1 + 1 + 0 = 0 2
1 + 1 + 1 = 0 3
0
1
3
0
1
3
0
1
0
0
1
0


a0
b0
s0
s'0
s1
a1
b1
s'1
a2
b2
s2
s'2
s3
a3
b3
s'3
s4
s'4

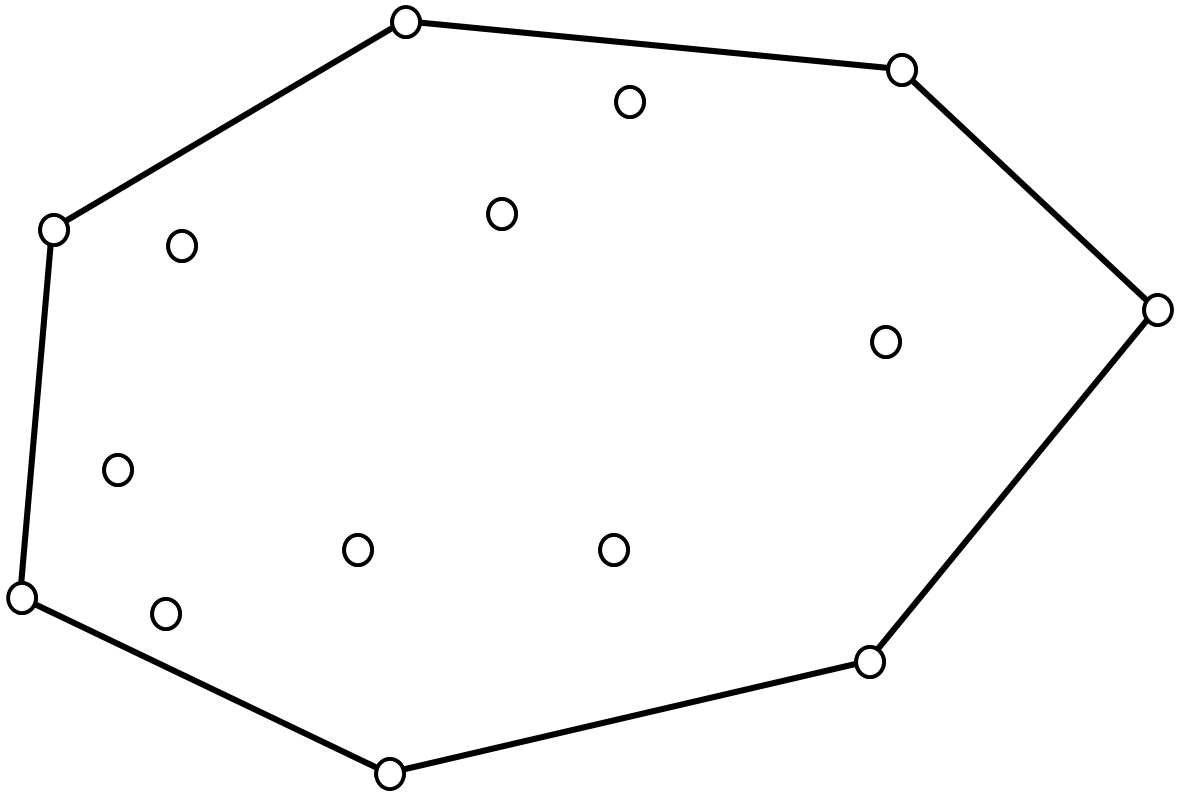
Calcul de l' enveloppe convexe d'un ensemble de points



Calcul de l' enveloppe convexe d'un ensemble de points

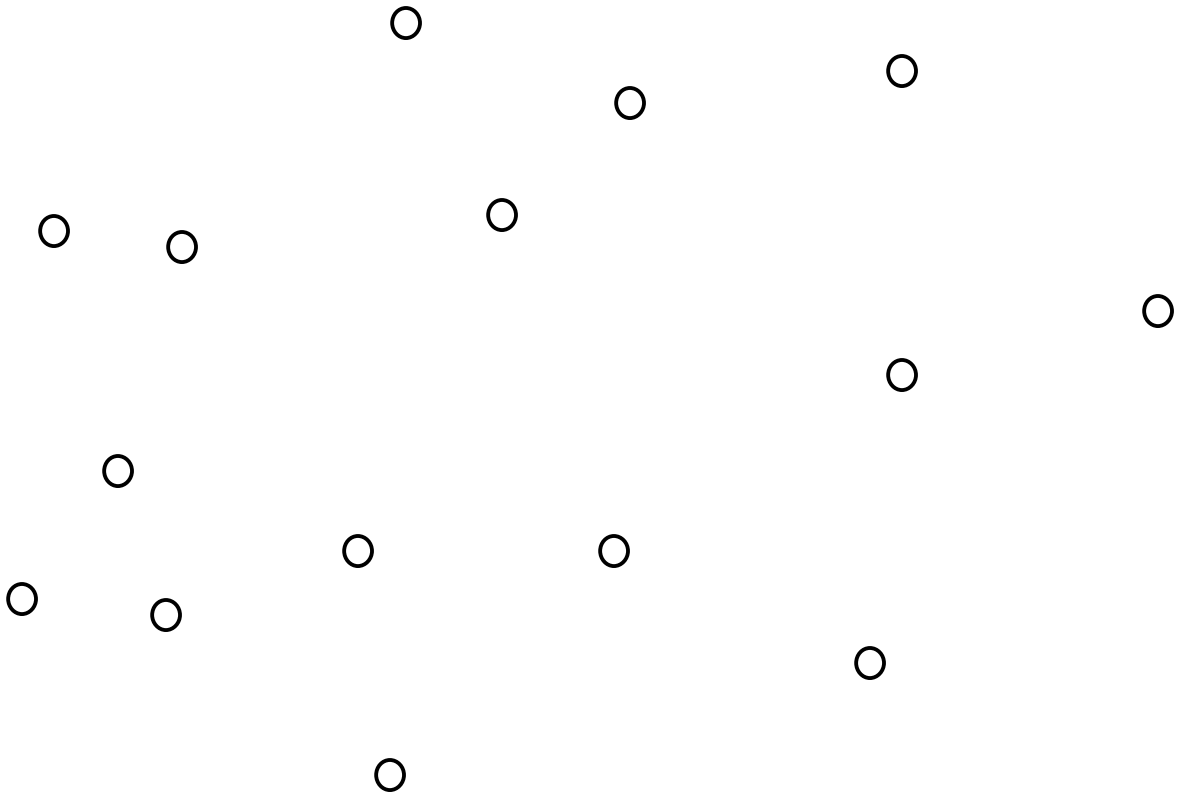


Calcul de l' enveloppe convexe d'un ensemble de points

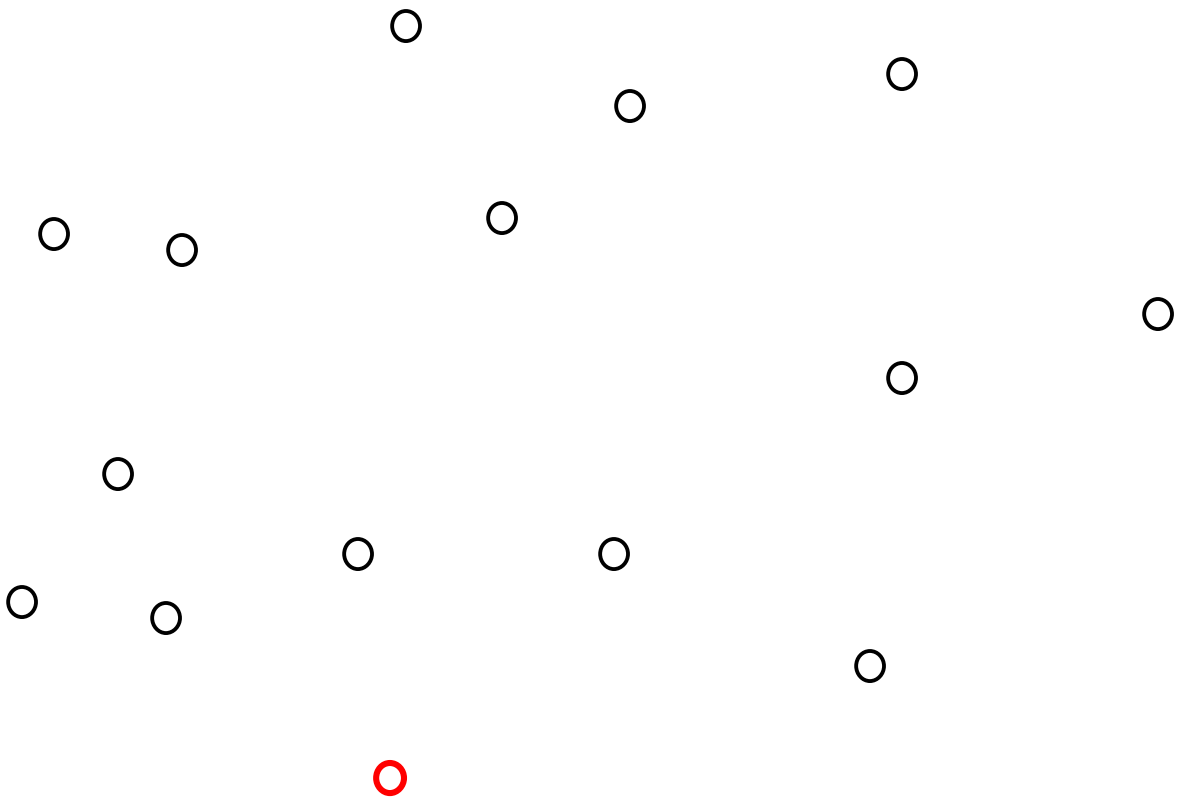


Un segment est dans l'enveloppe convexe
si et seulement si tous les autres points sont du même côté

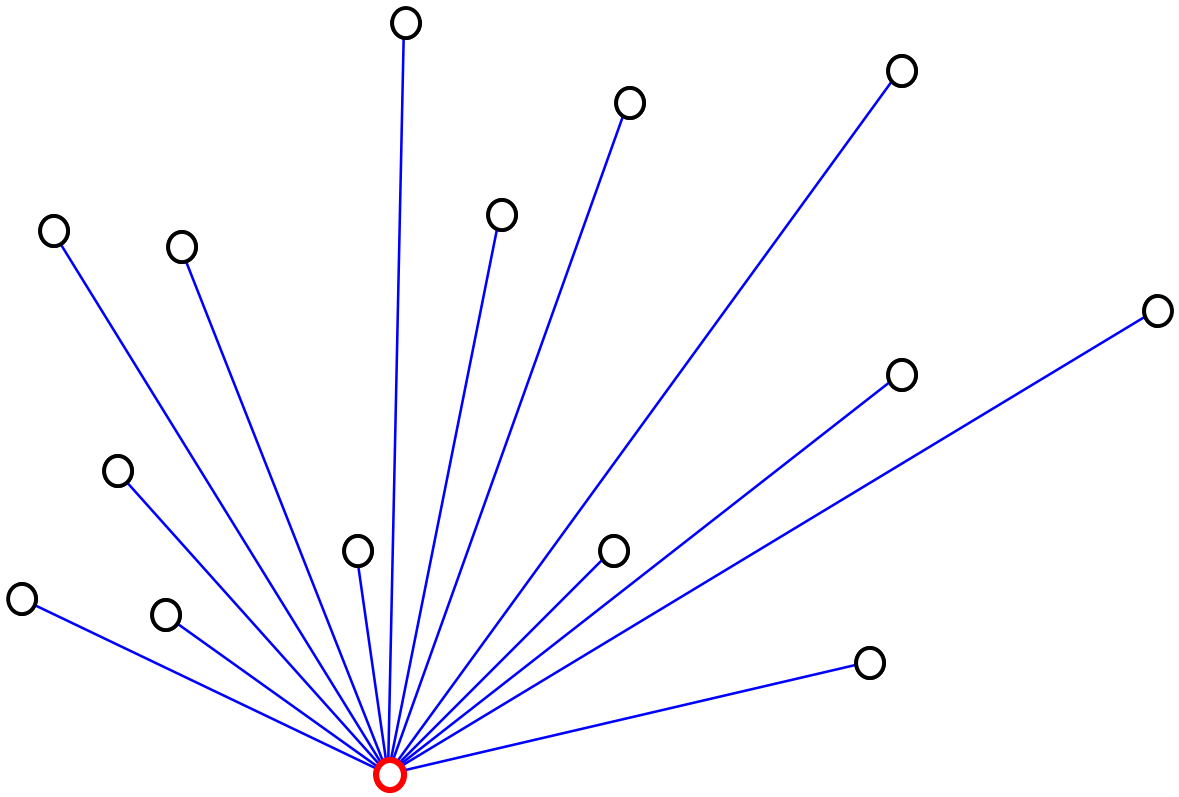
Point le plus bas, calcul des angles



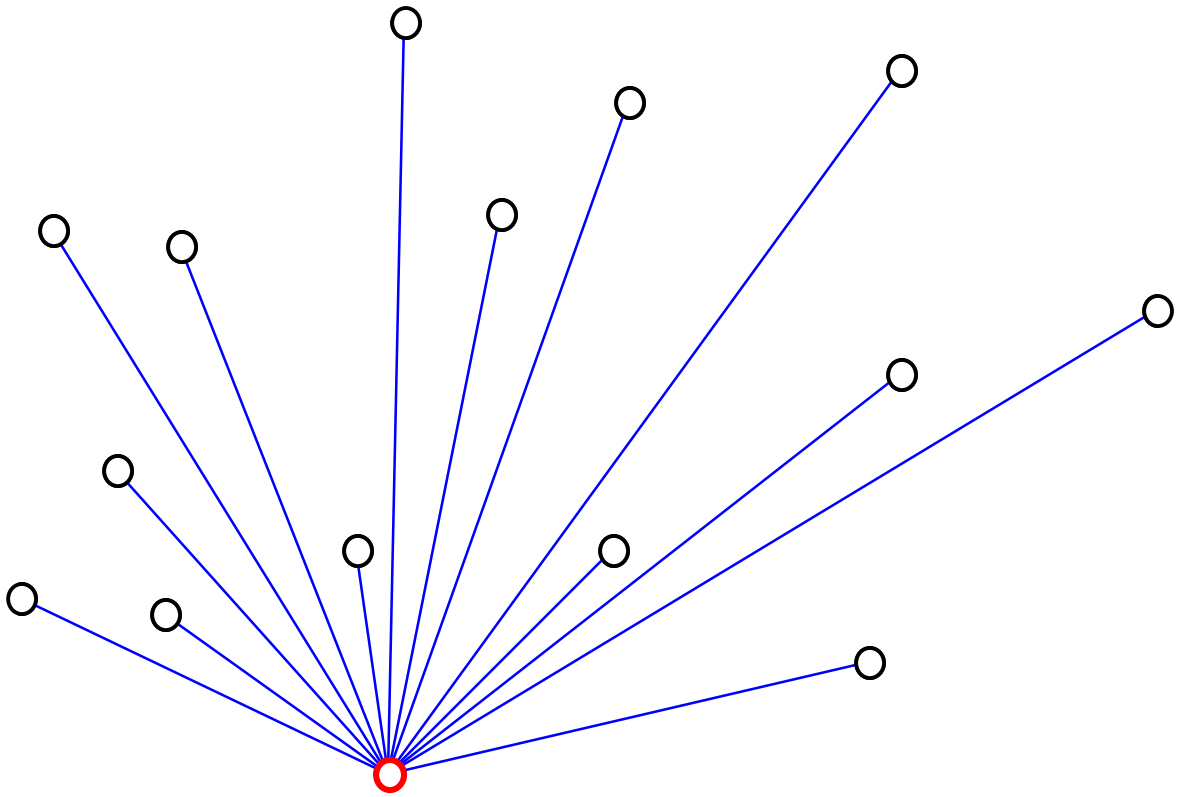
Point le plus bas, calcul des angles



Point le plus bas, calcul des angles

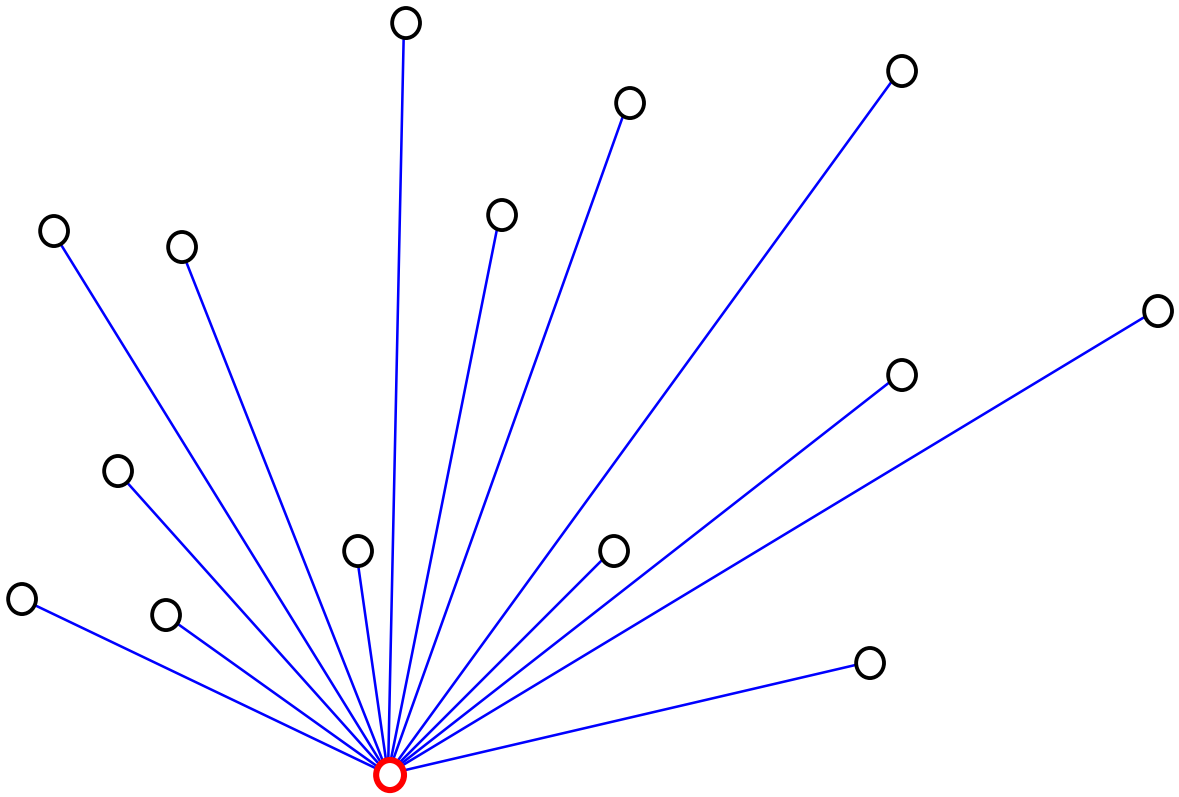


Point le plus bas, calcul des angles

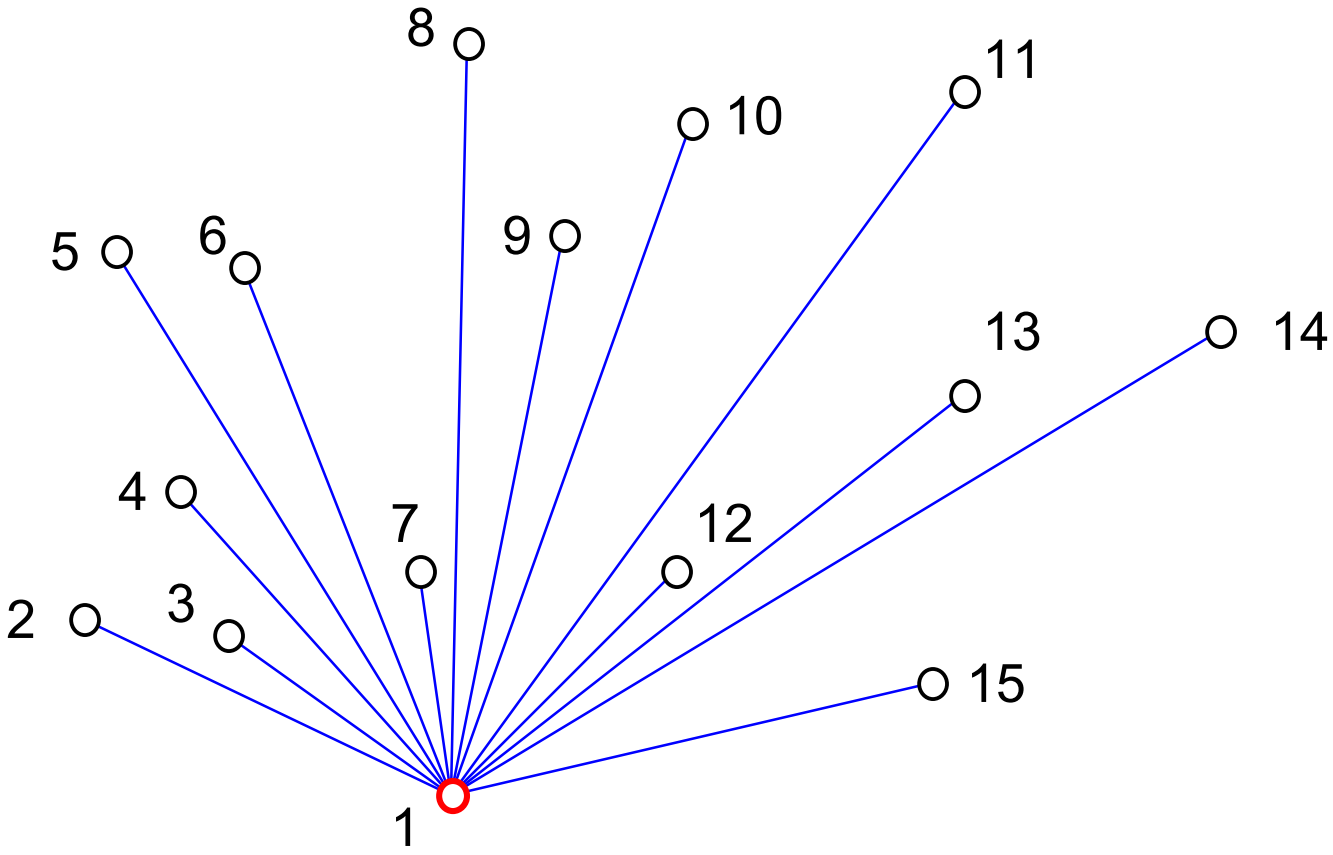


coût : O(n)

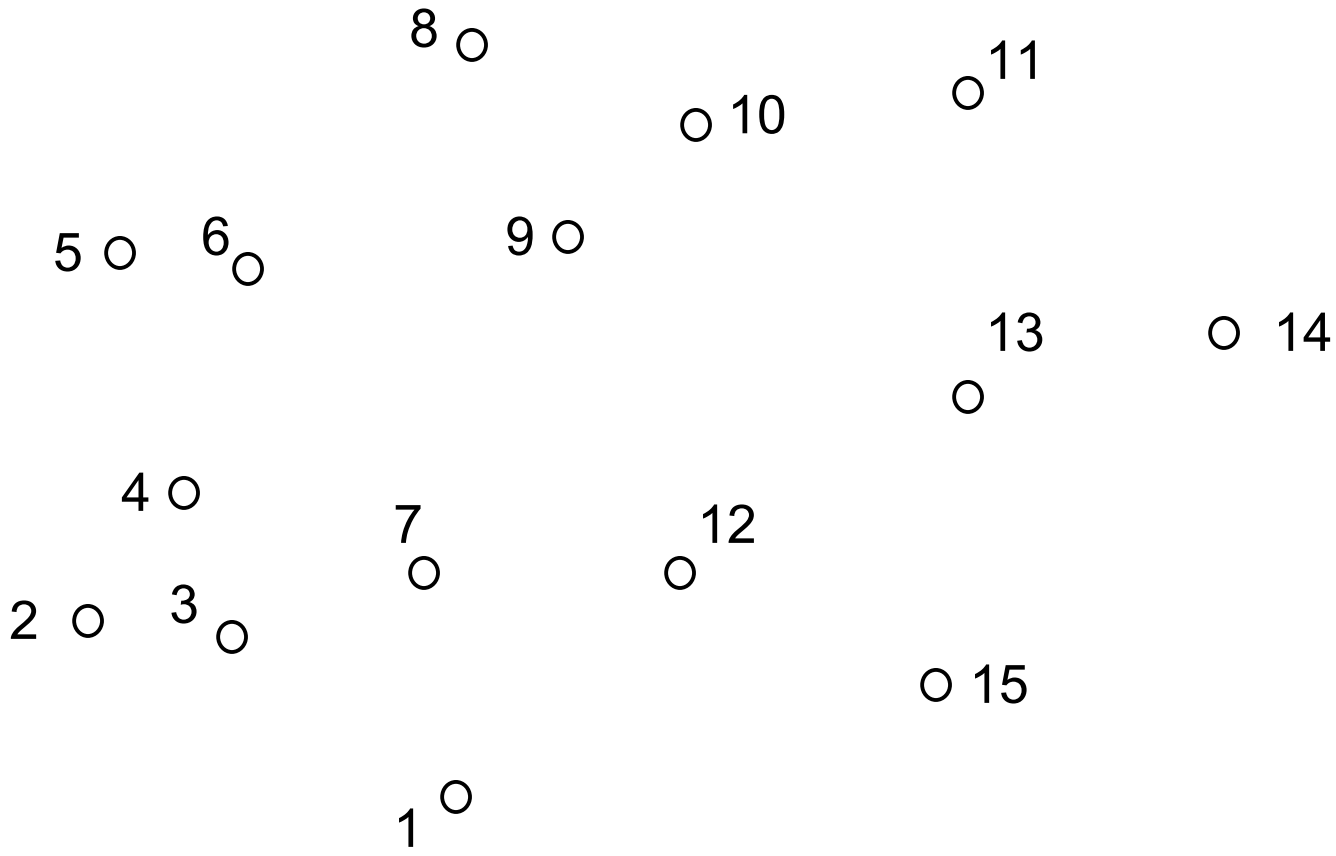
Tri des angles, numérotation



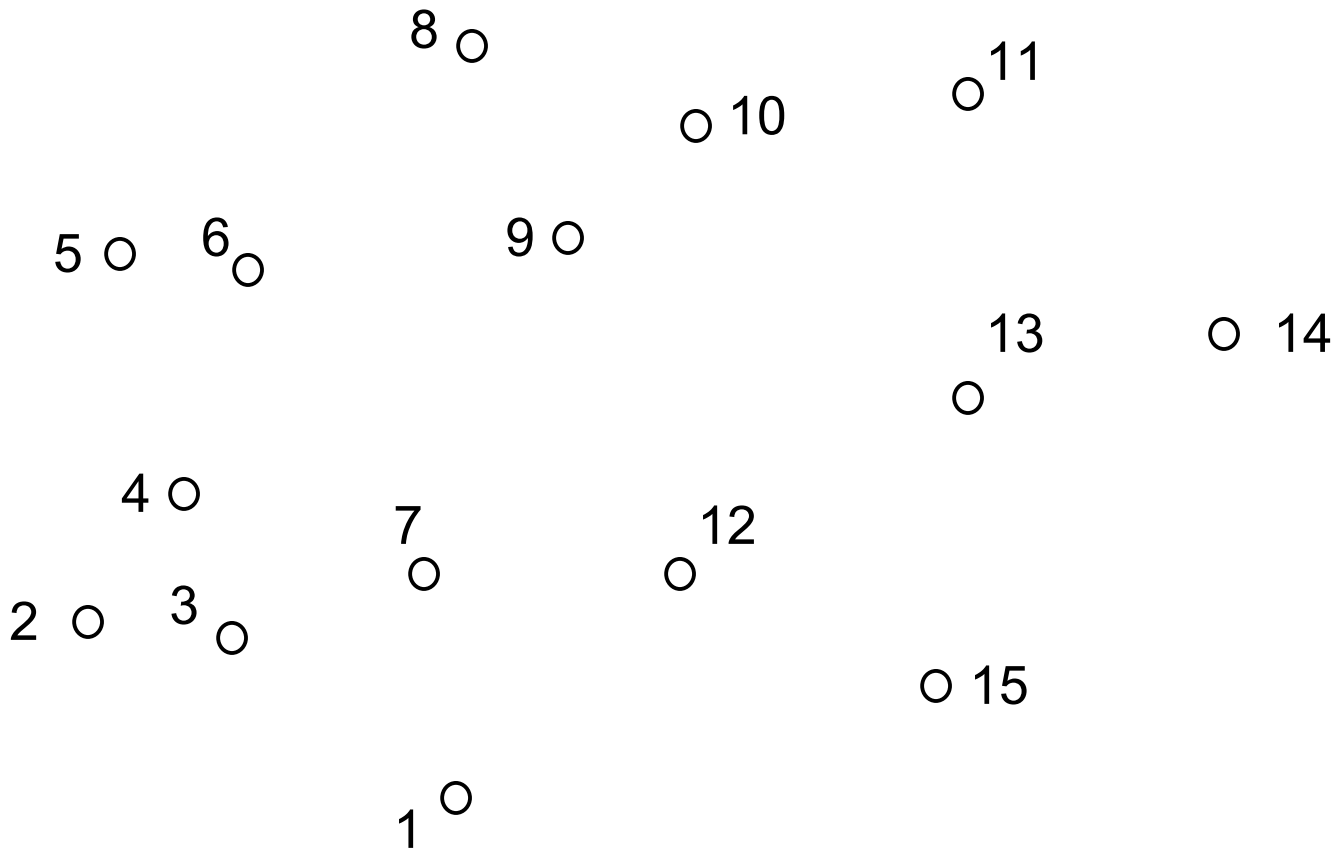
Tri des angles, numérotation



Tri des angles, numérotation

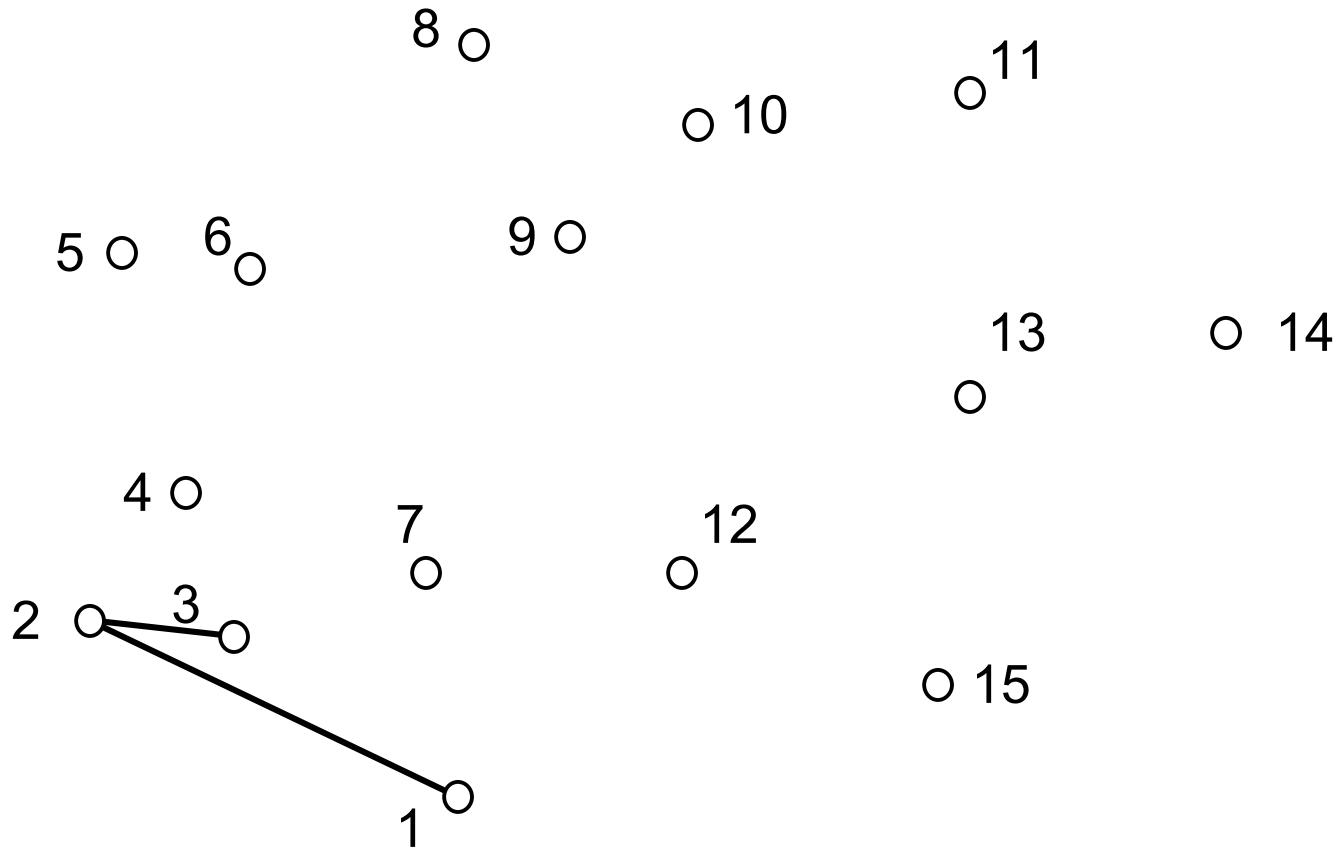


Tri des angles, numérotation

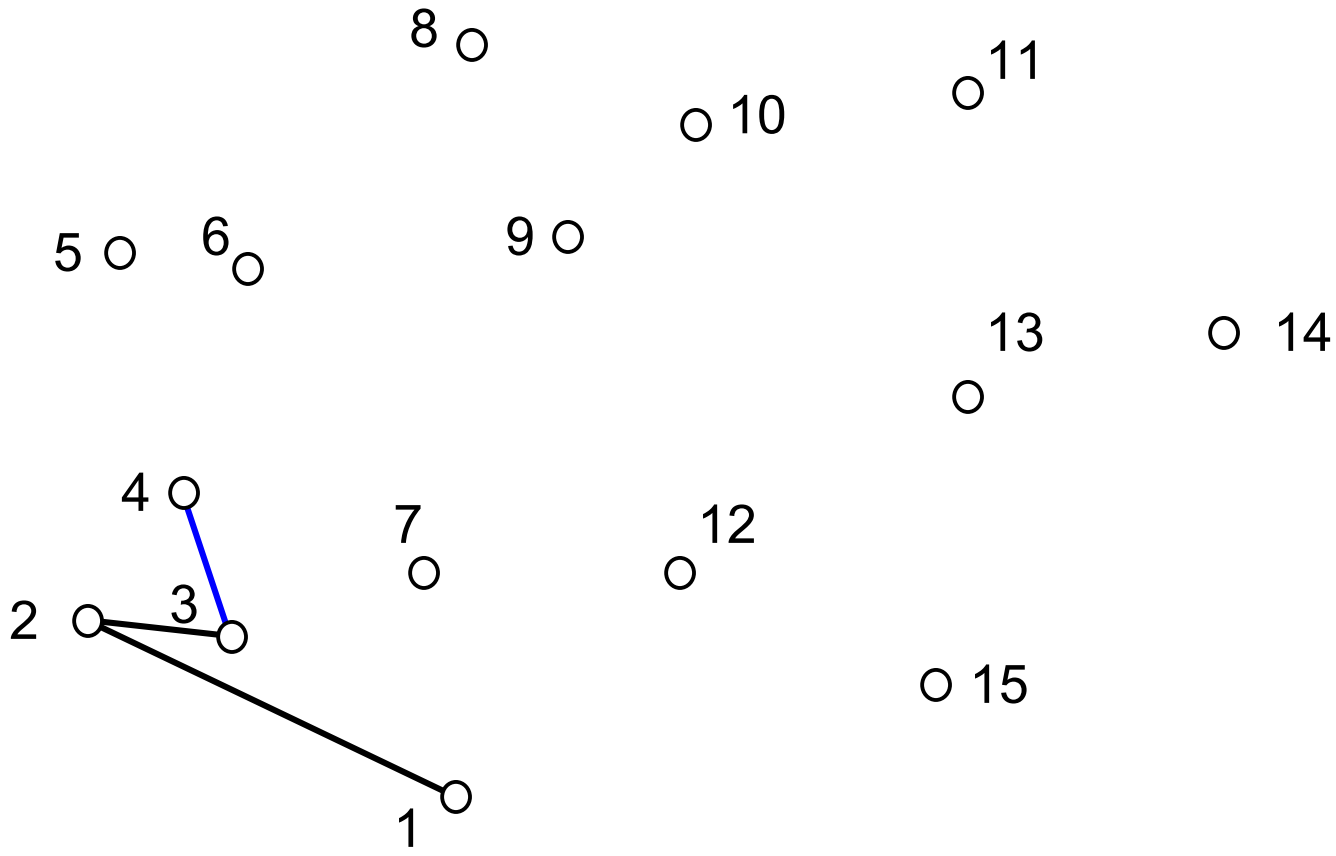


coût : O(n logn)

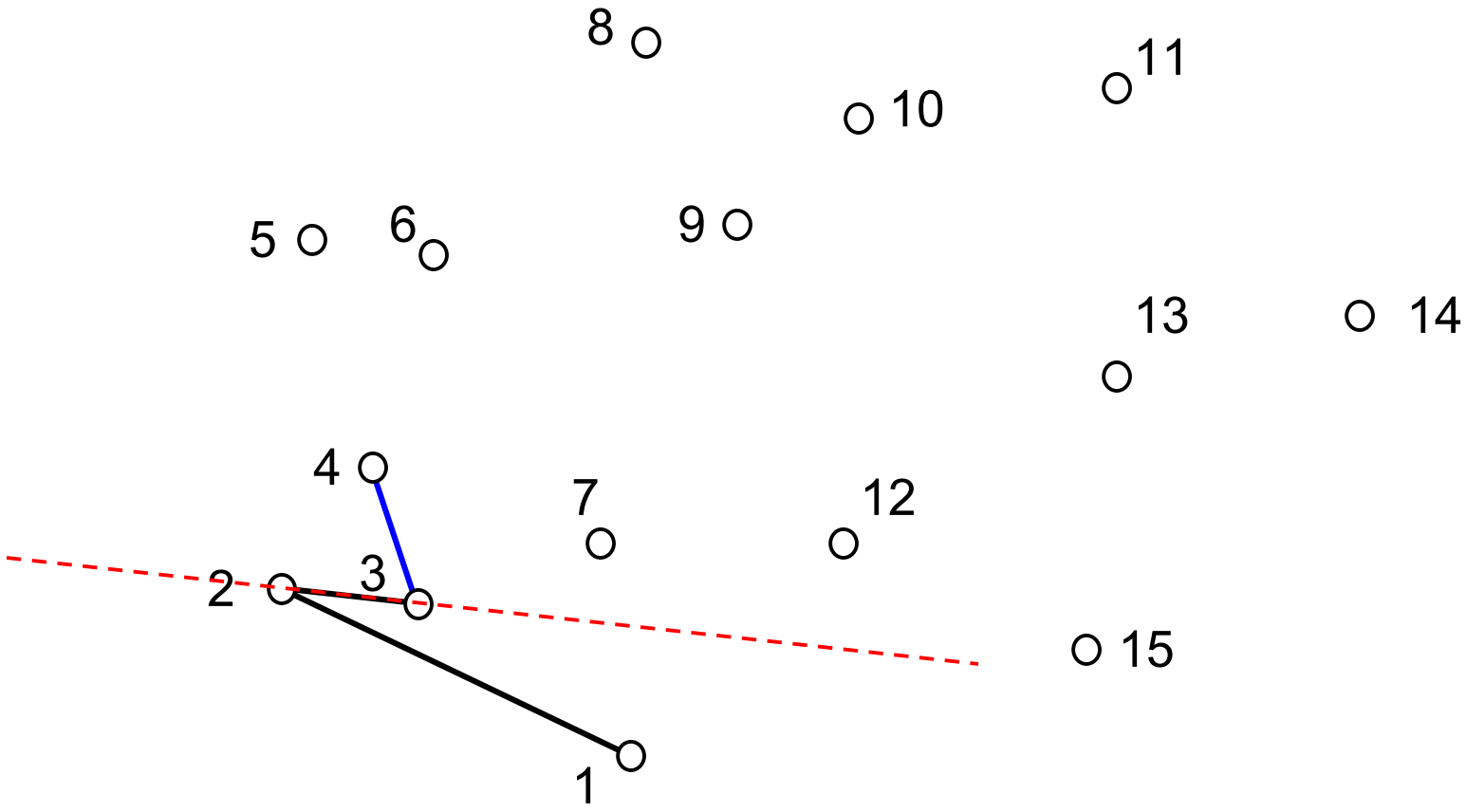
3. Début de l'enveloppe: 1,2,3, partez !



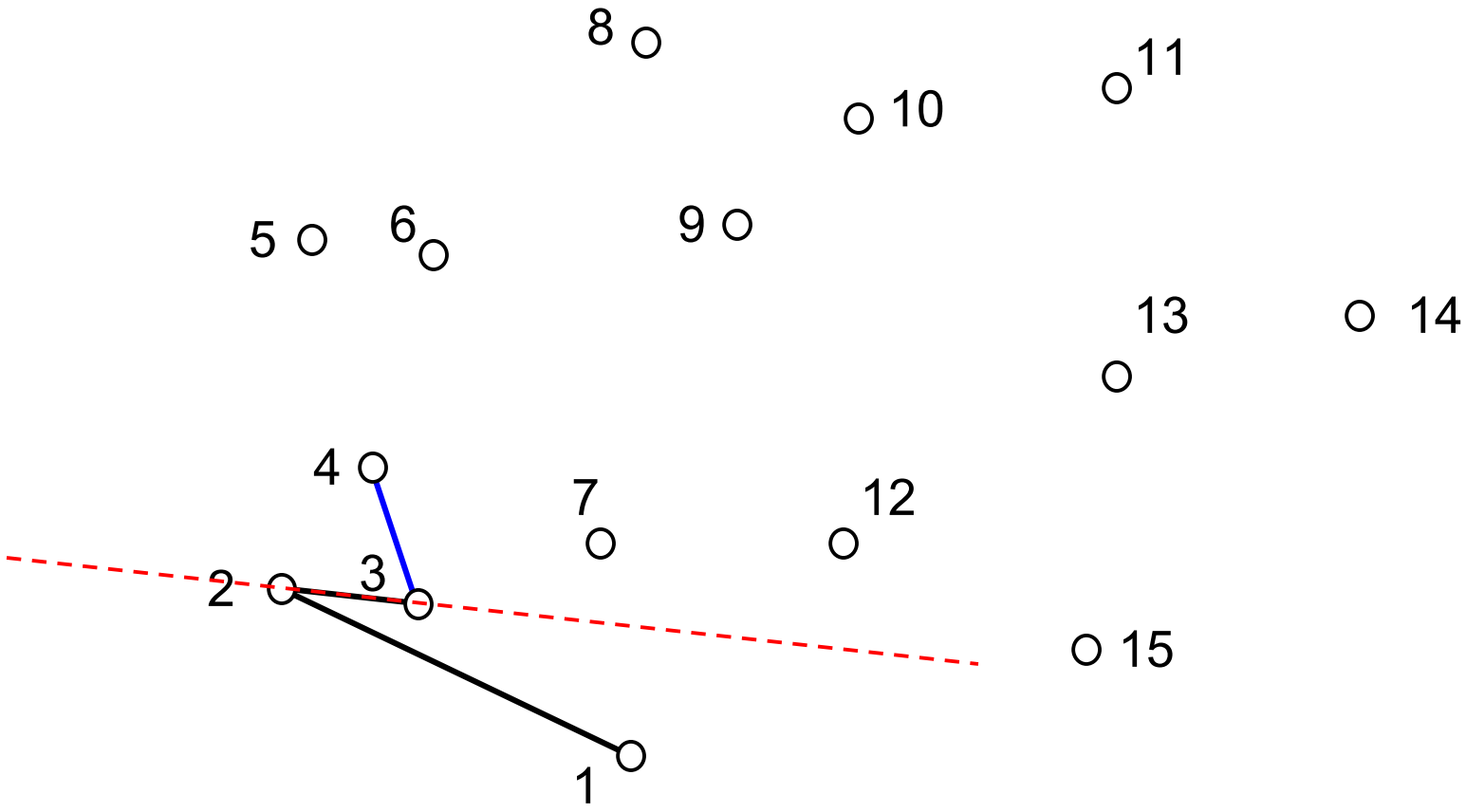
Ajout de 4



Ajout de 4

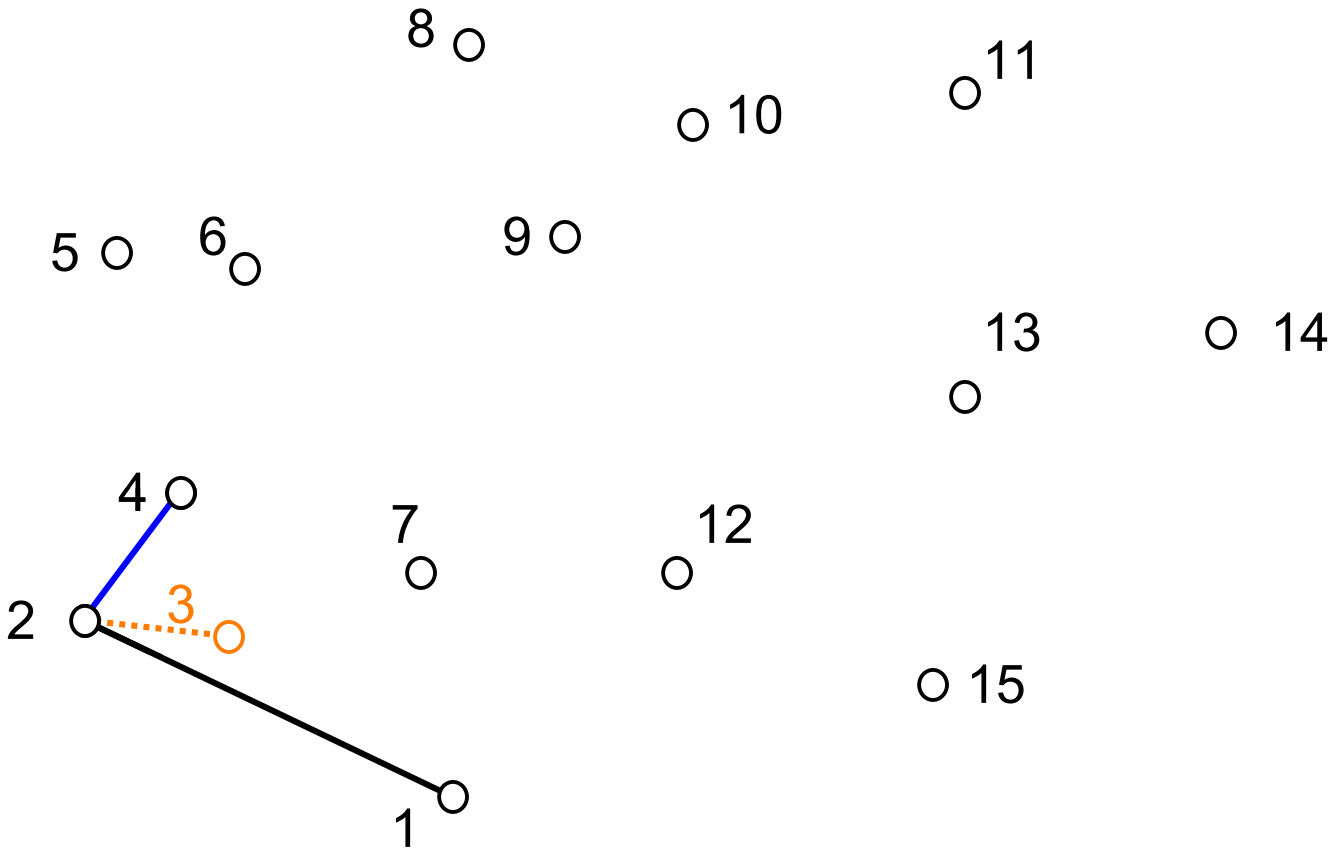


Ajout de 4



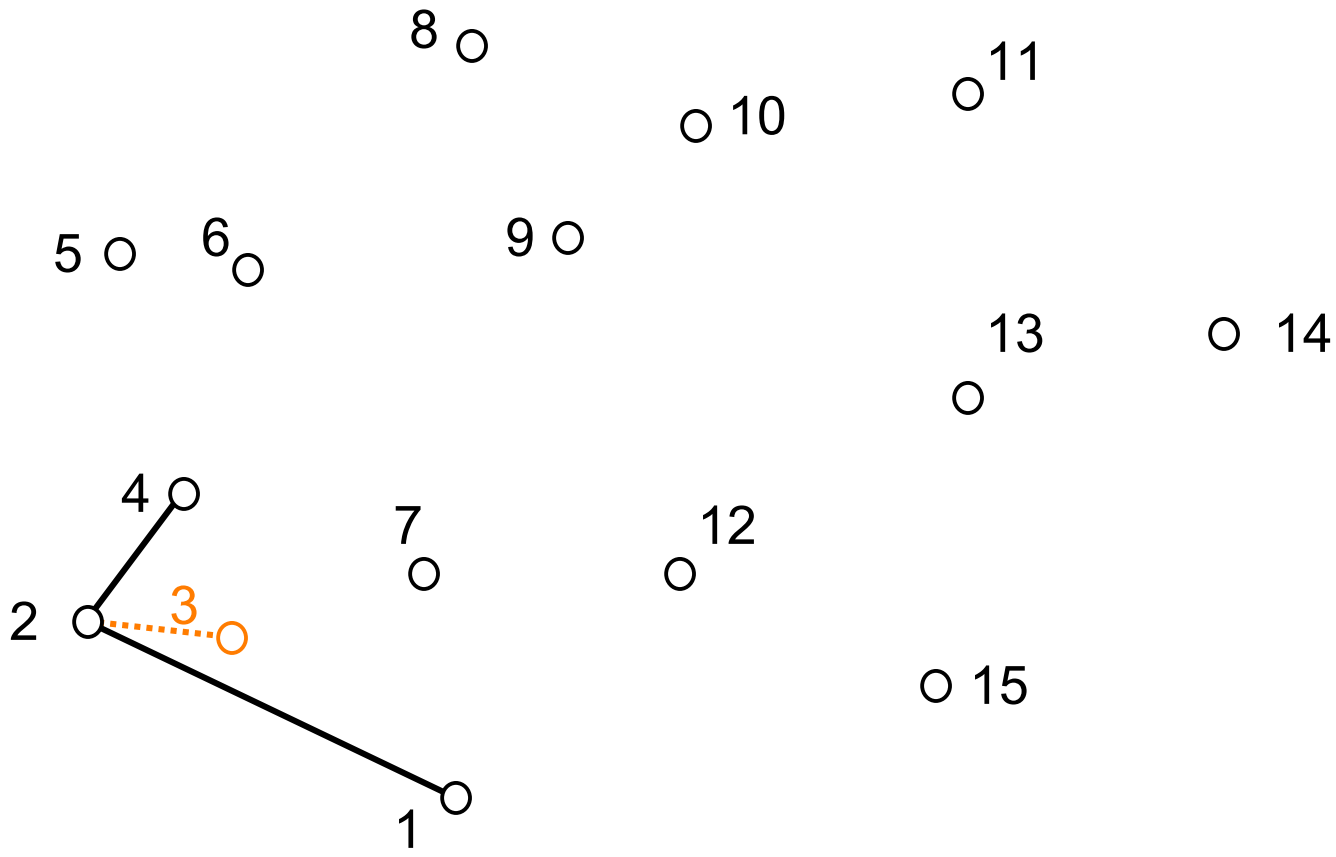
Suppression de 3, car 1 et 4 de chaque côté de 3-2

Ajout de 4



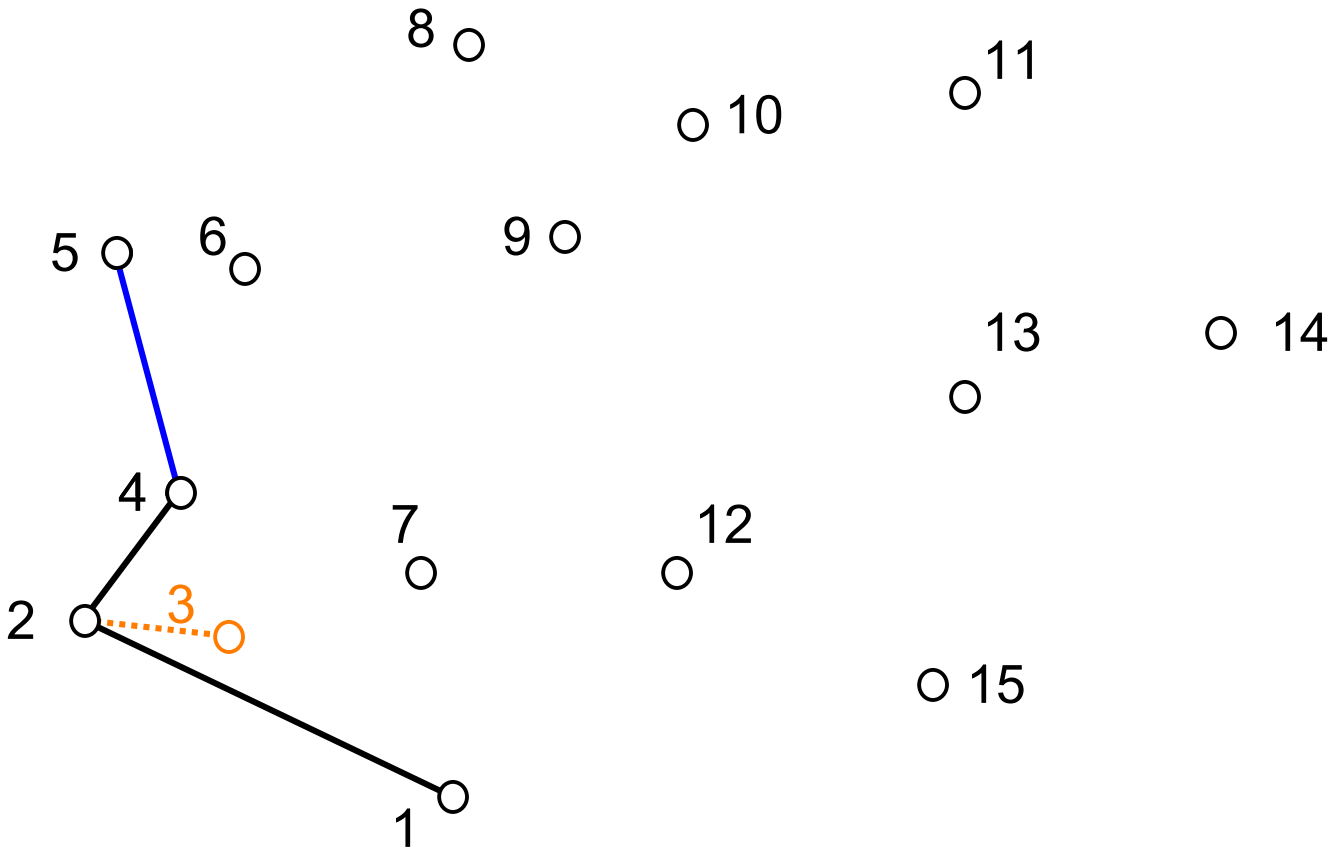
Suppression de 3, car 1 et 4 de chaque côté de 3-2

Ajout de 4



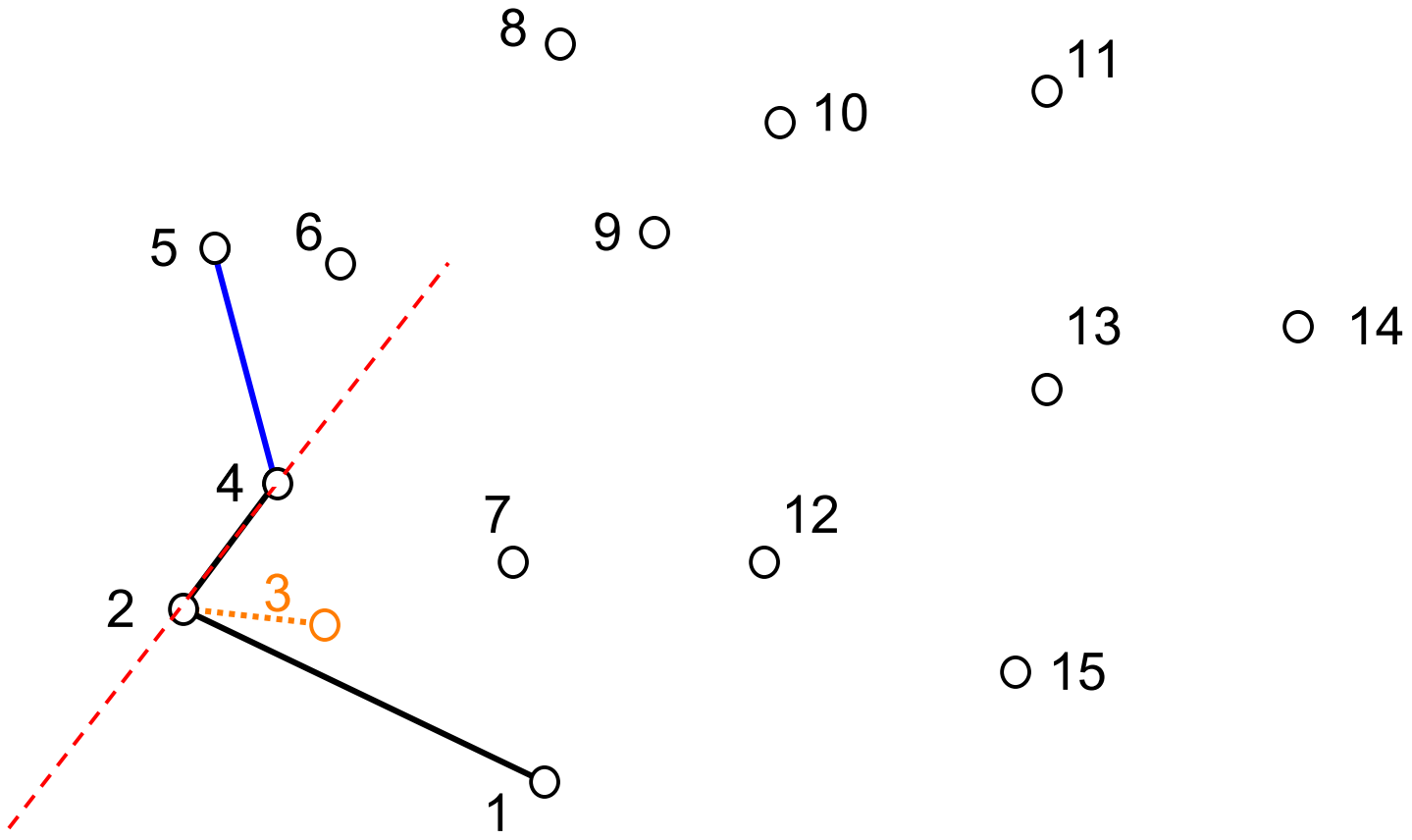
Plus rien à supprimer

Ajout de 5



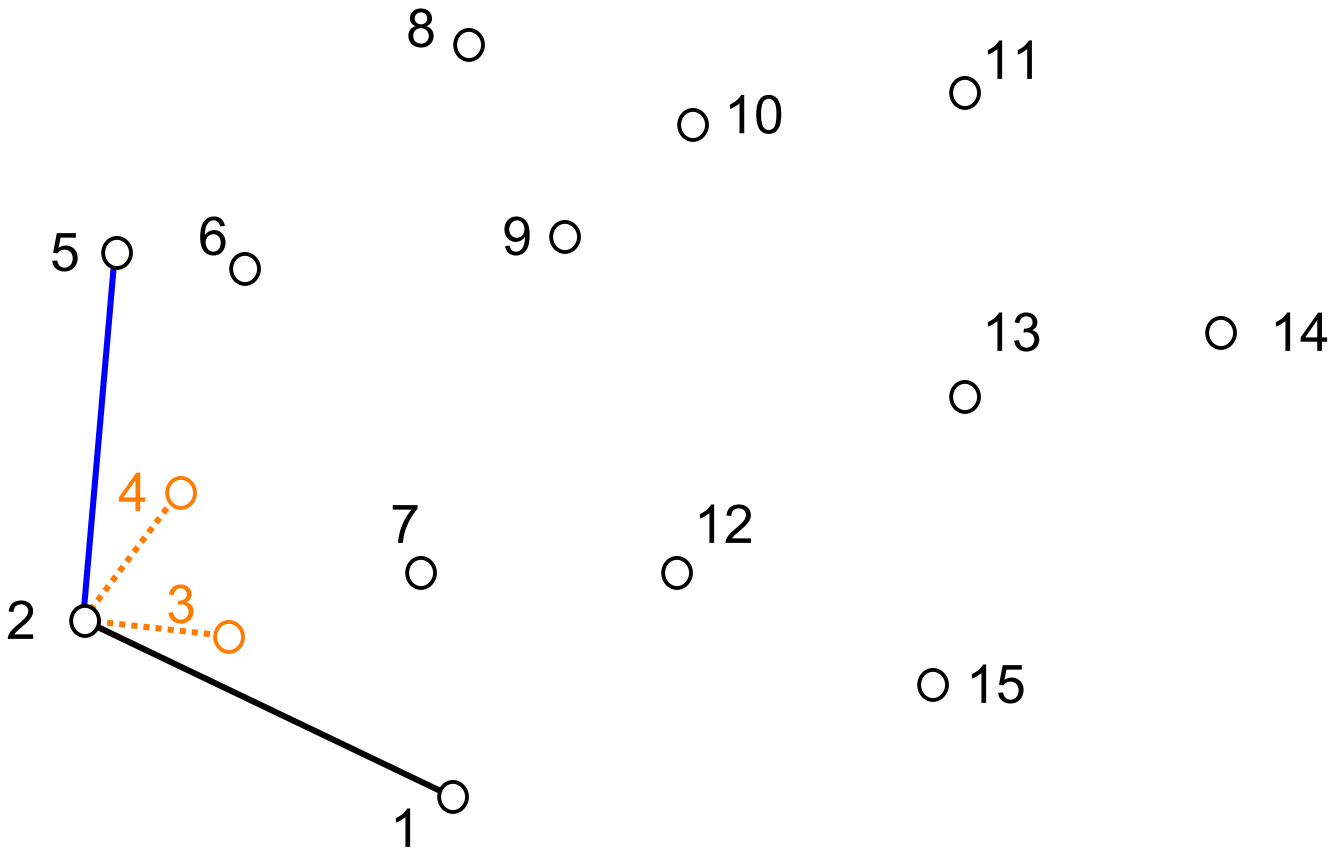
Suppression de 4, car 1 et 5 de chaque côté de 4-2

Ajout de 5

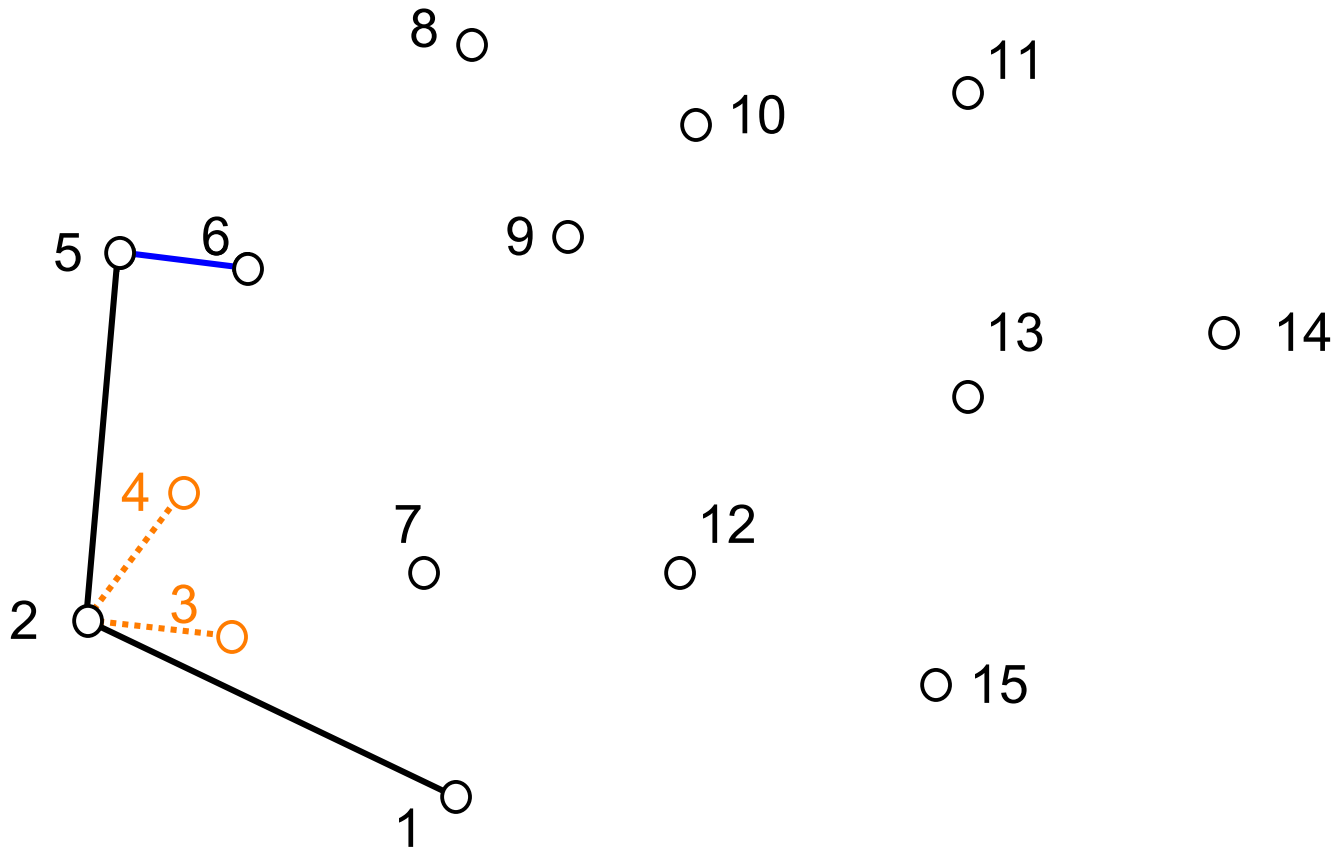


Suppression de 4, car 1 et 5 de chaque côté de 4-2

Ajout de 5

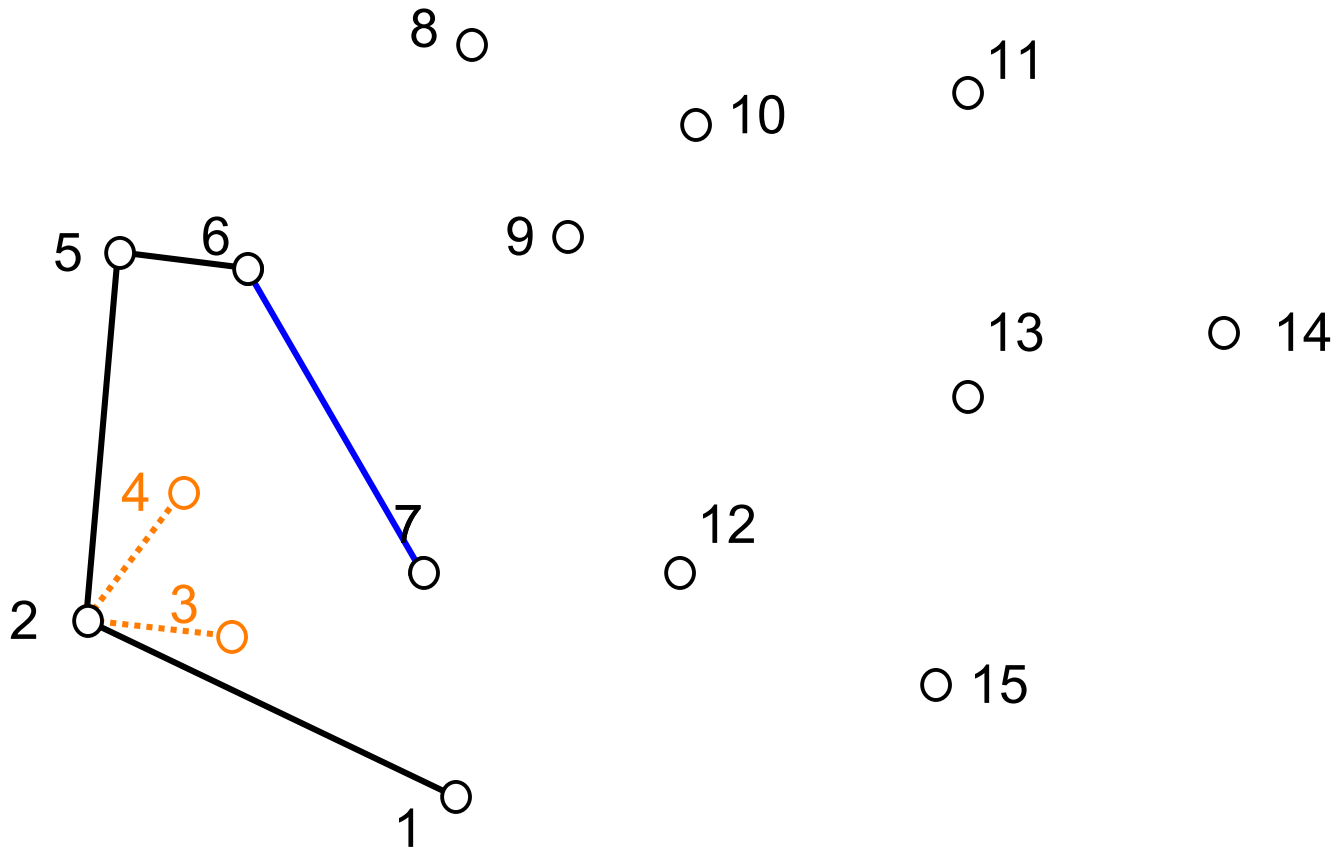


Ajout de 6



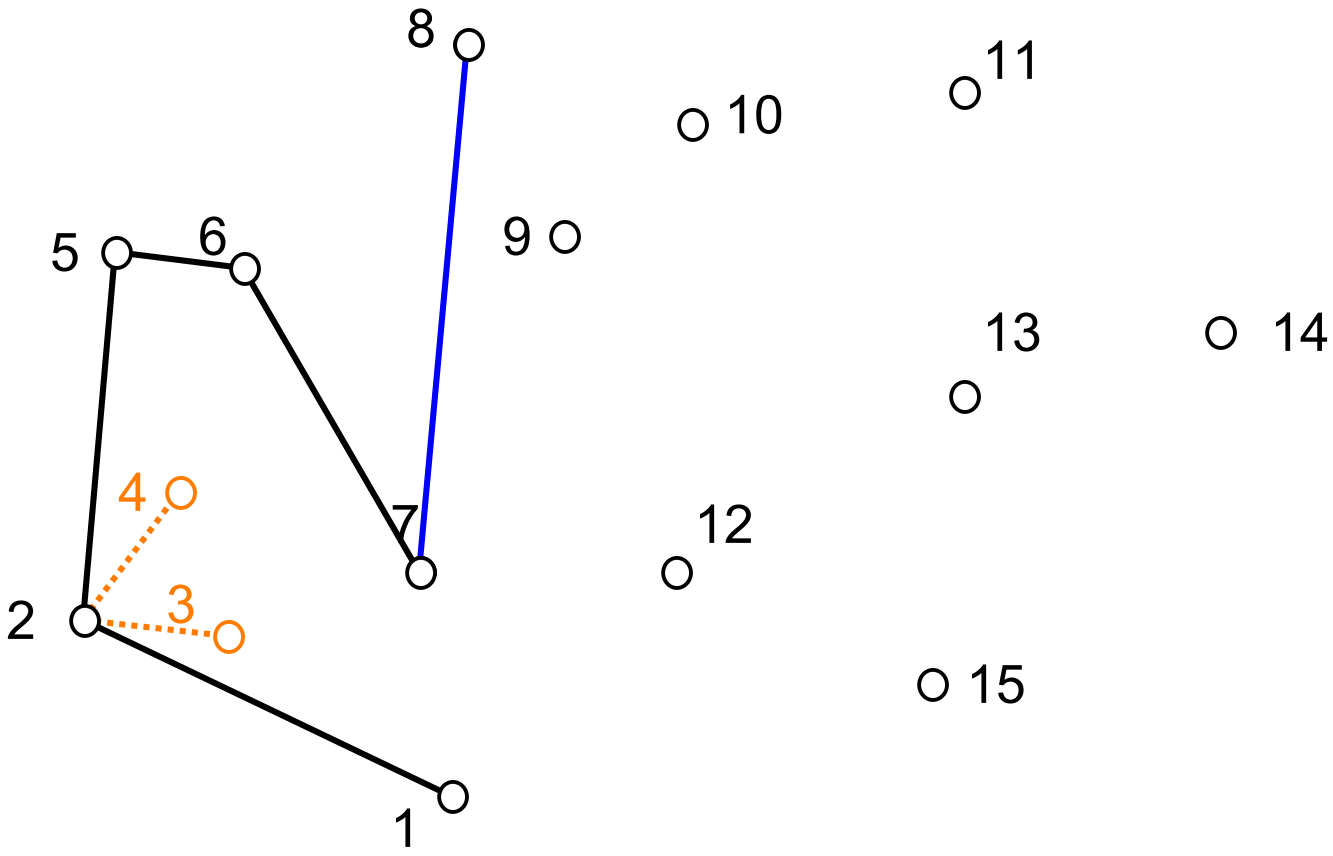
Rien à supprimer

Ajout de 7



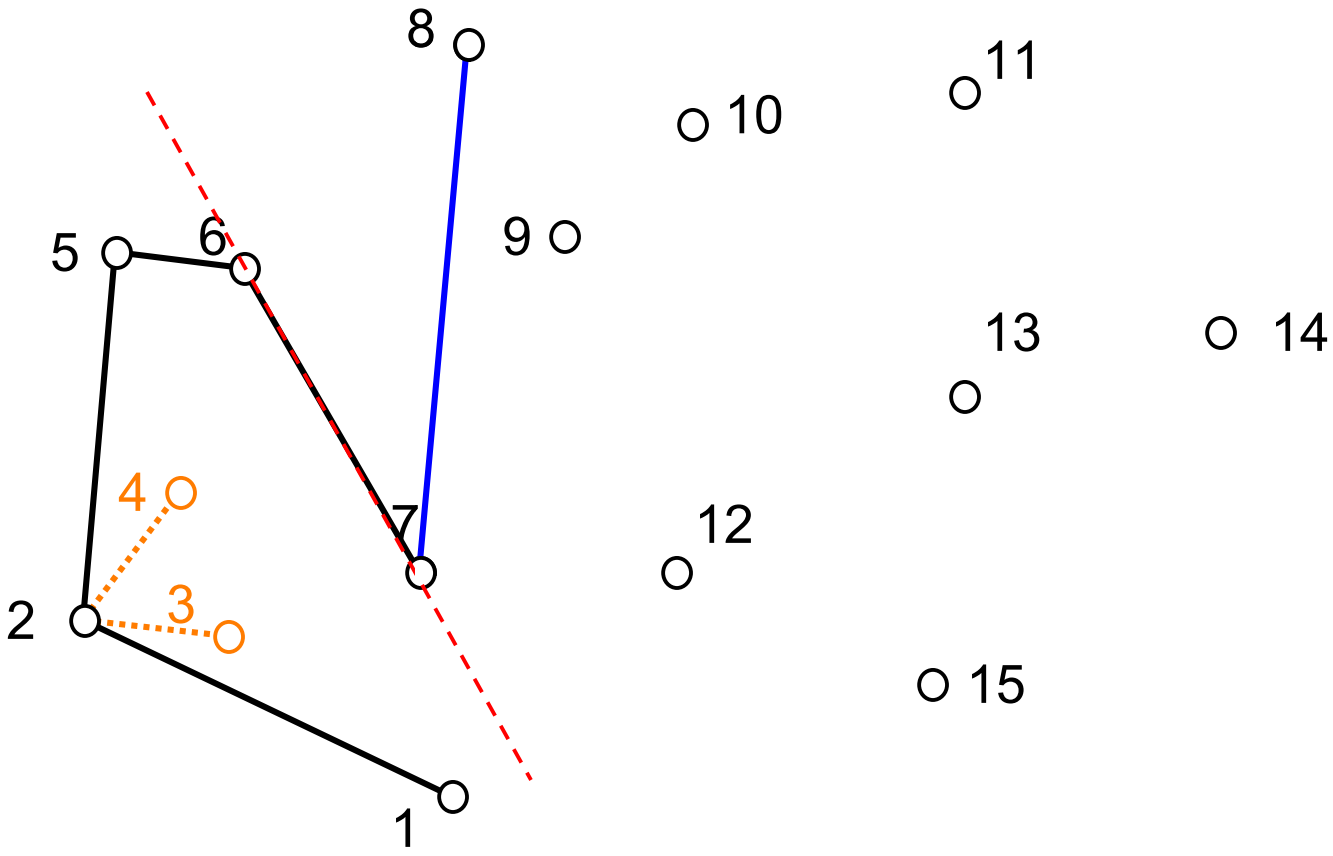
Rien à supprimer

Ajout de 8



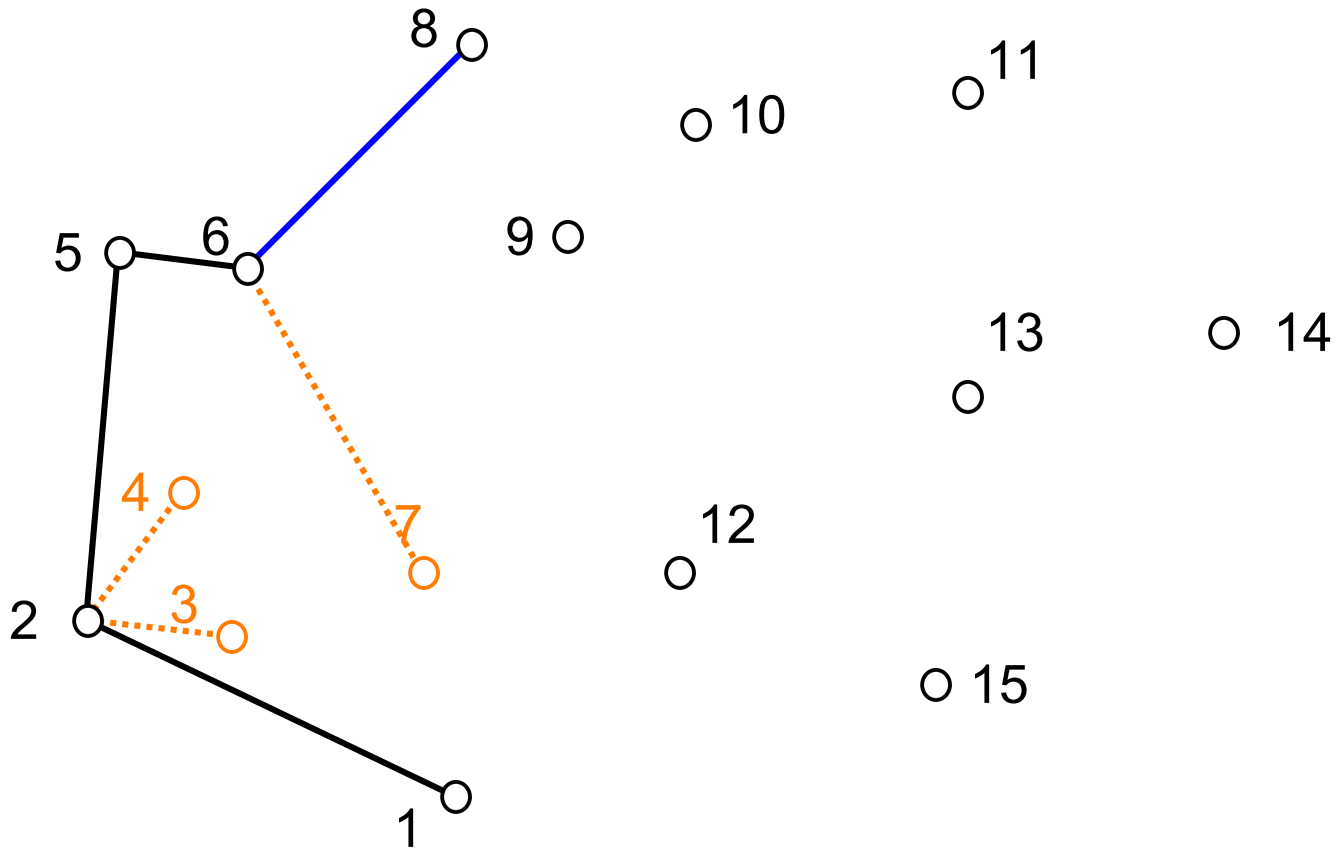
Suppression de 7 car 1 et 8 de chaque côté de 7-6

Ajout de 8



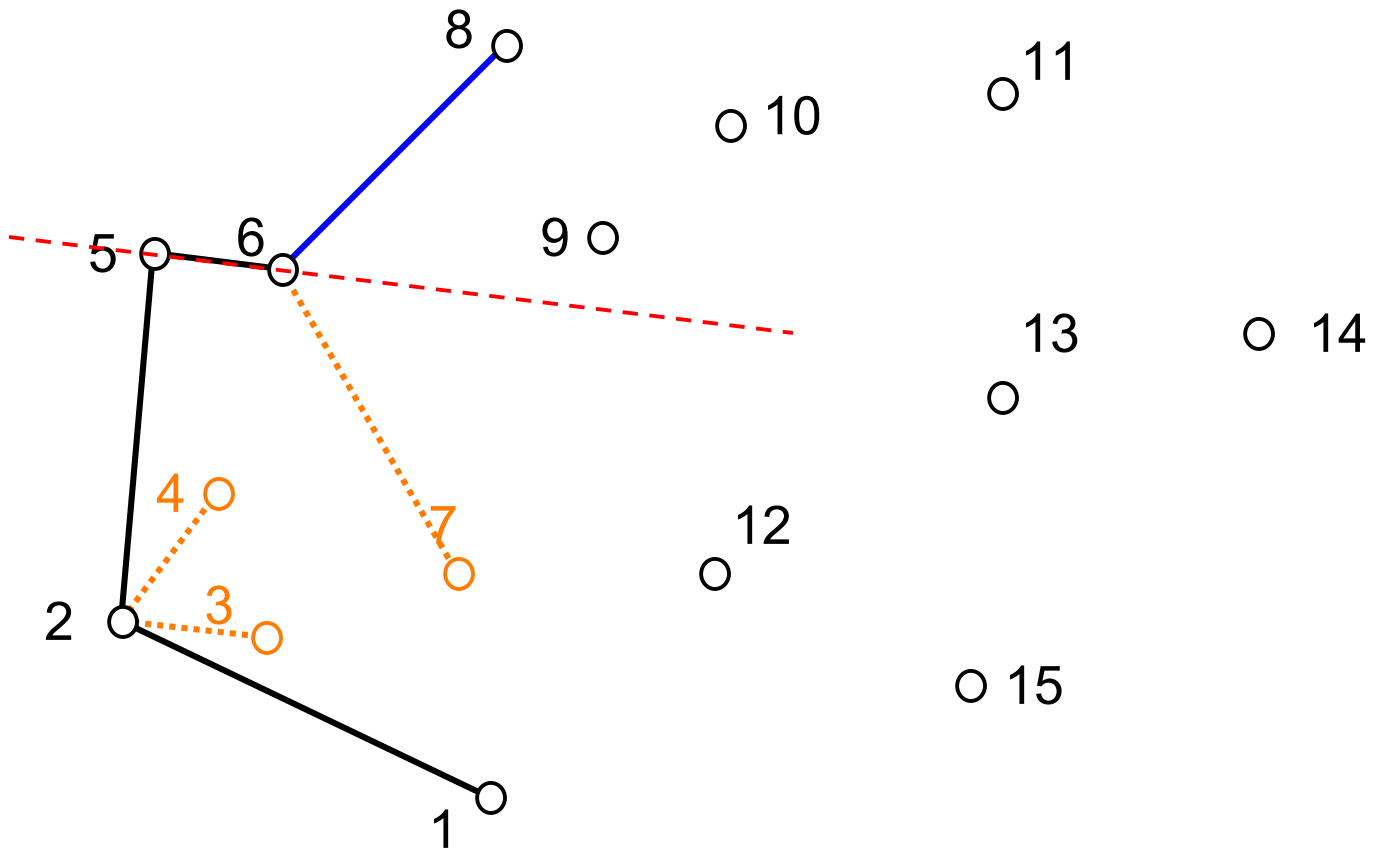
Suppression de 7 car 1 et 8 de chaque côté de 7-6

Ajout de 8



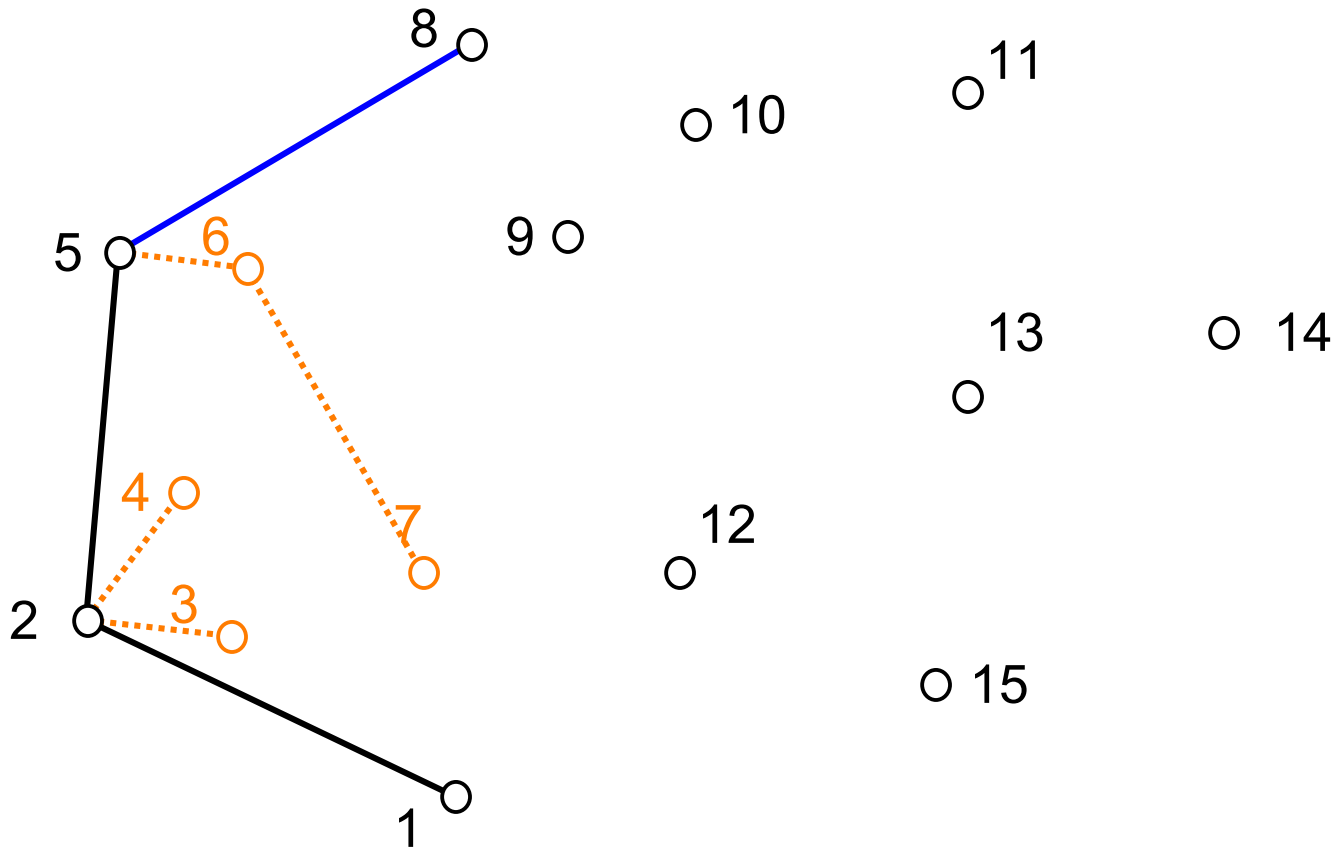
Suppression de 6 car 1 et 8 de chaque côté de 6-5

Ajout de 8



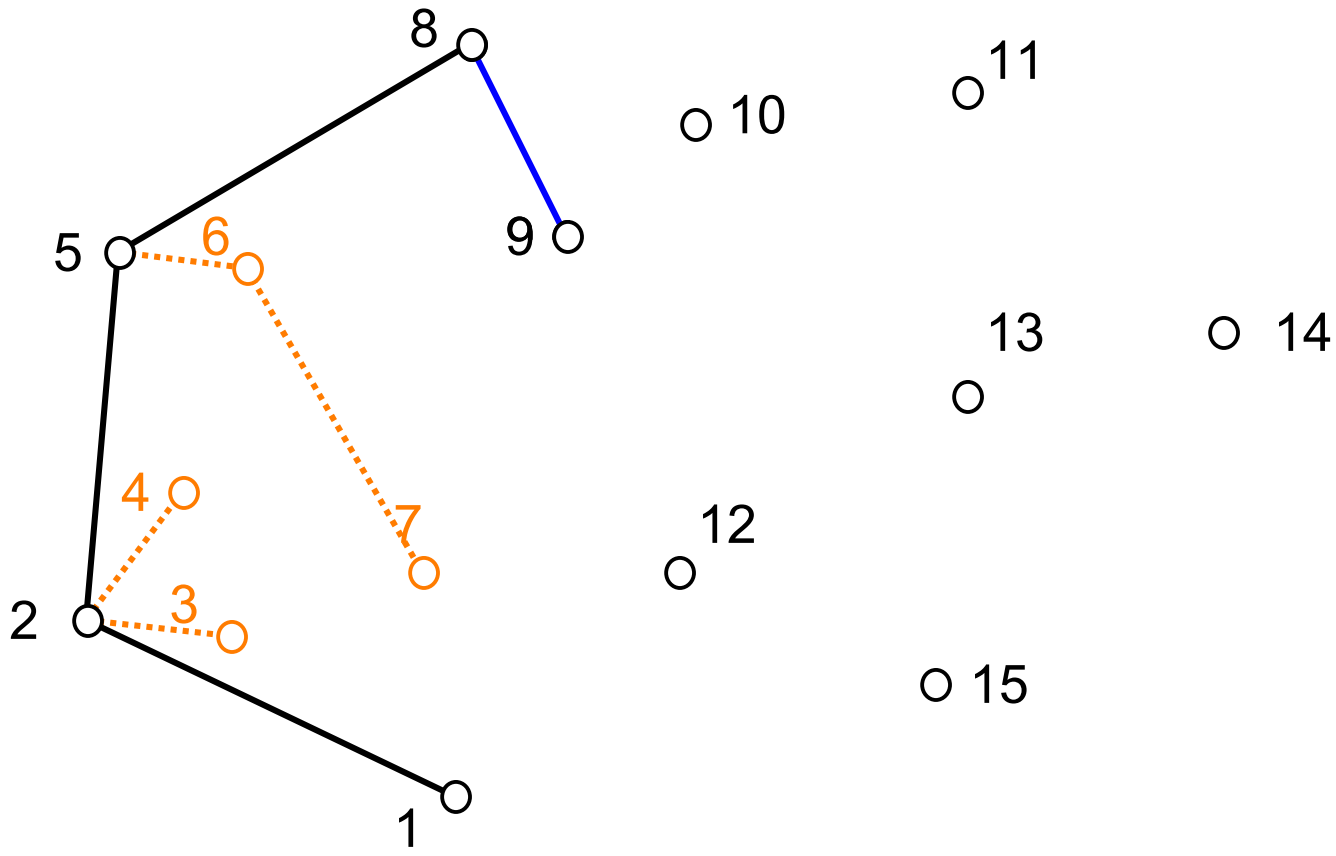
Suppression de 6 car 1 et 8 de chaque côté de 6-5

Ajout de 8



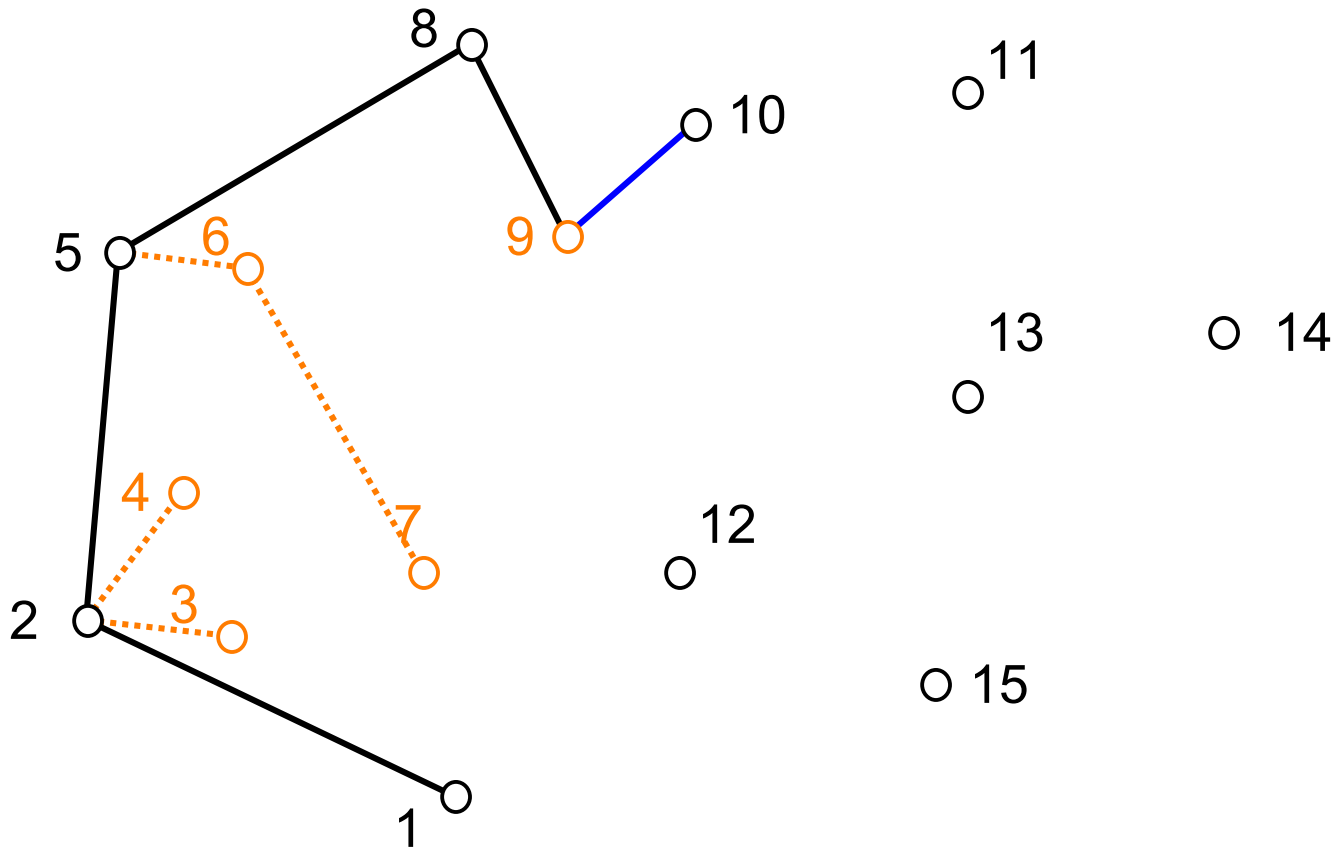
Plus rien à supprimer

Ajout de 9



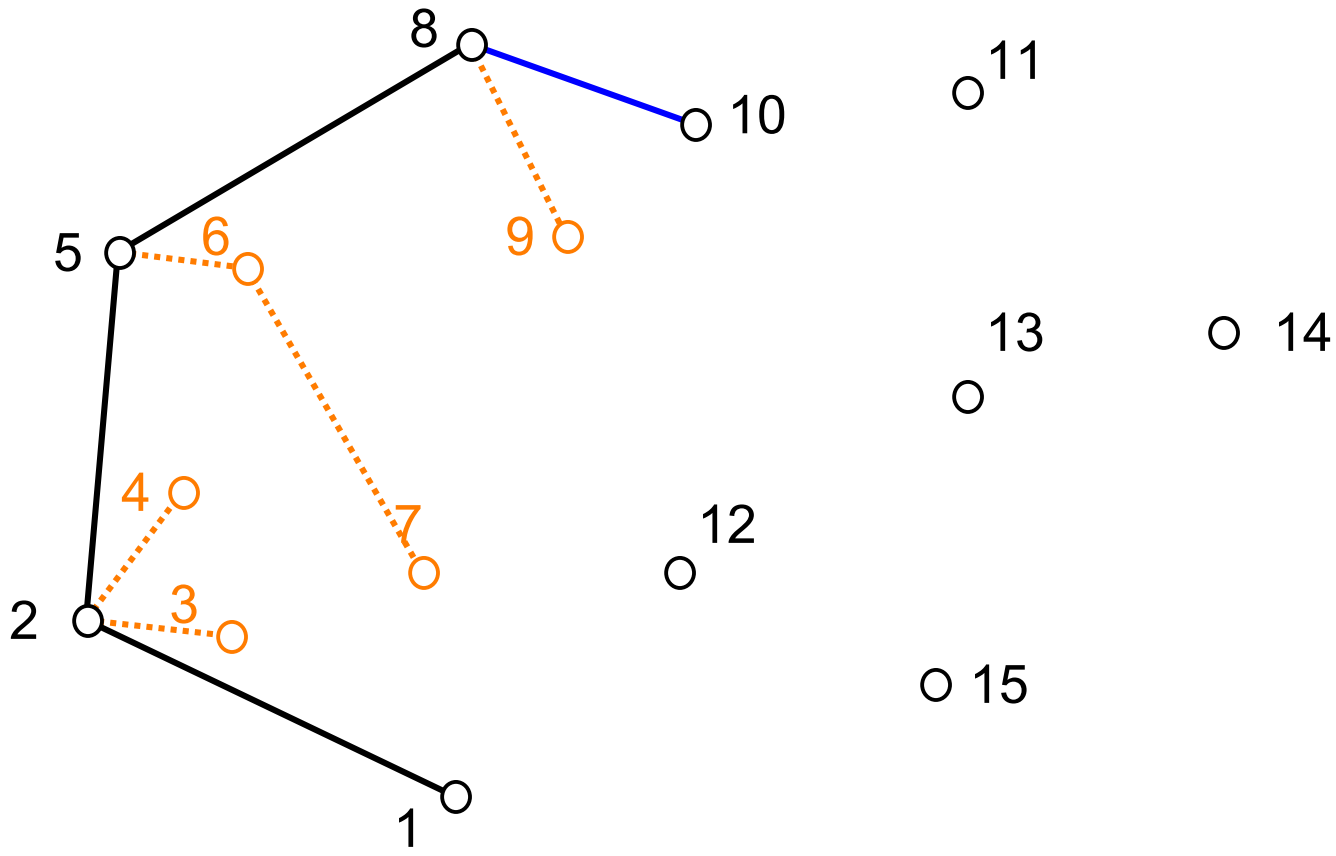
Rien à supprimer

Ajout de 10



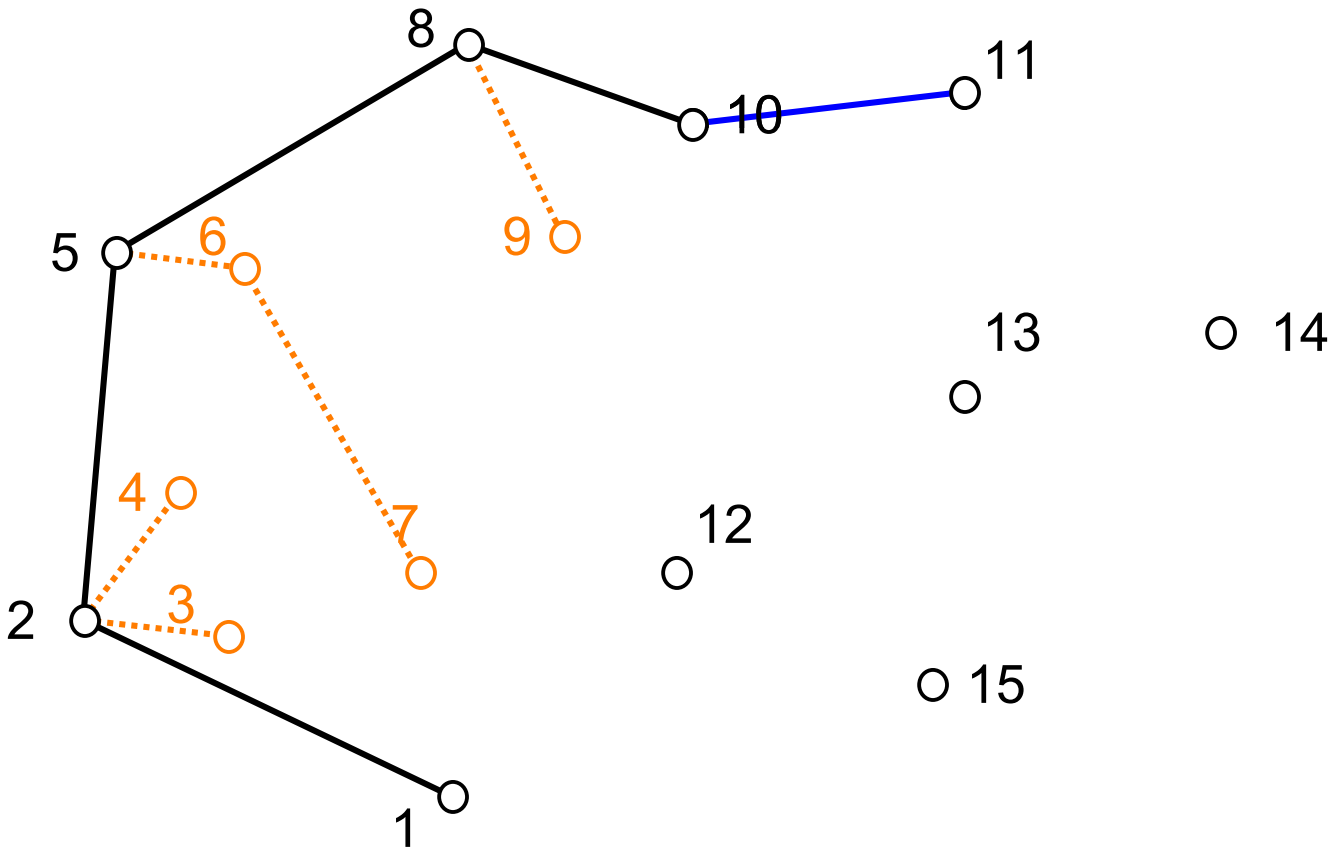
Suppression de 9 car 1 et 10 de chaque côté de 9-8

Ajout de 10

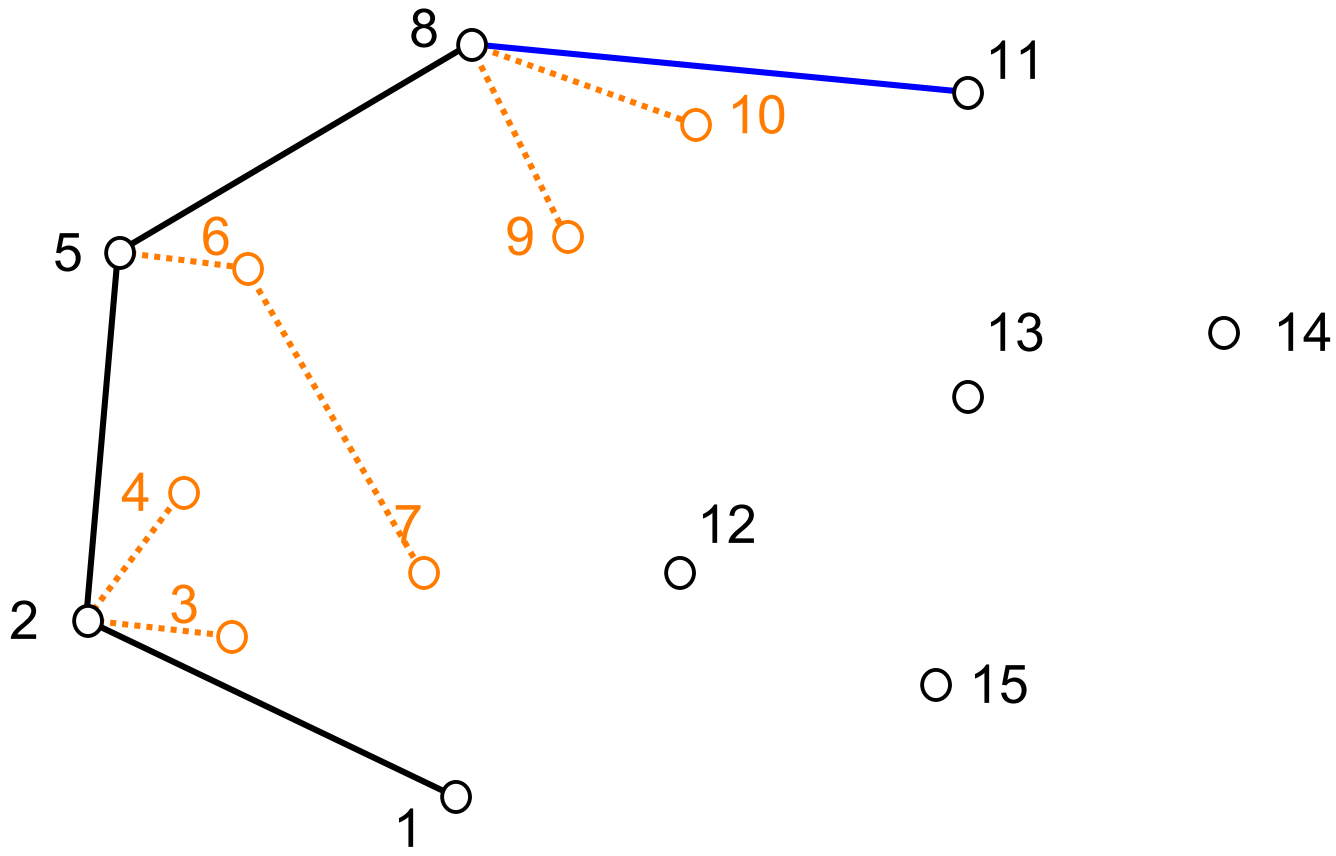


Suppression de 9 car 1 et 10 de chaque côté de 9-8

Ajout de 11

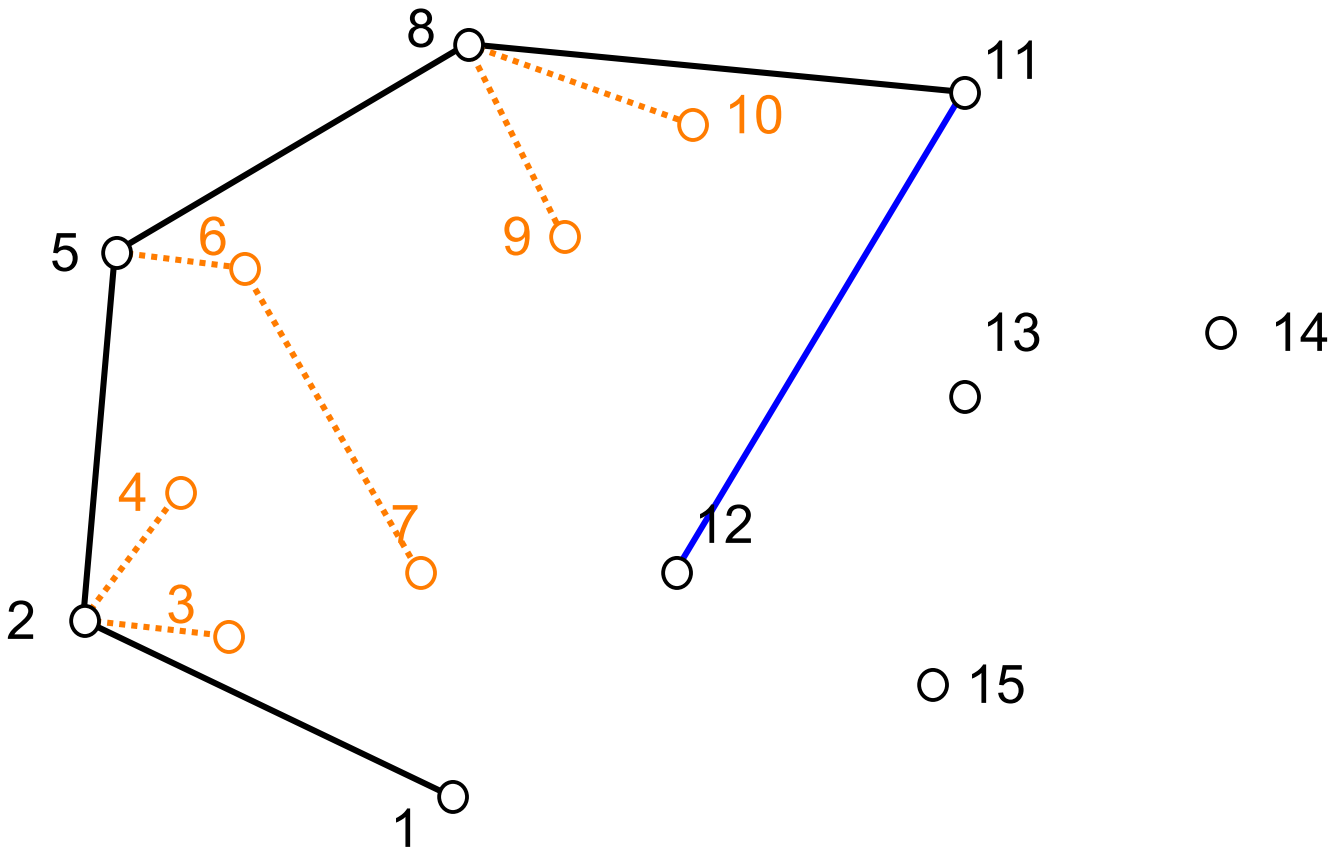


Ajout de 11



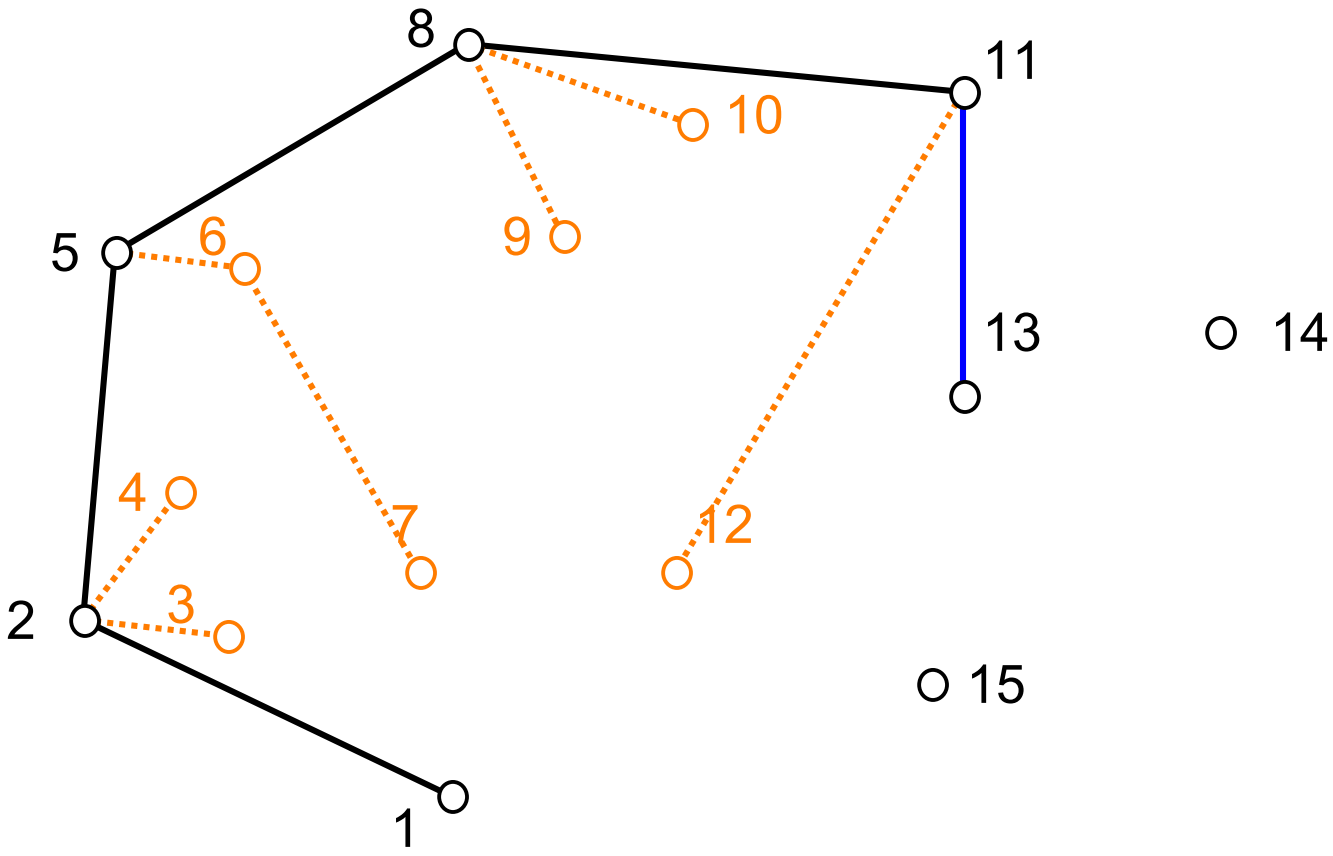
Suppression de 10 car 1 et 11 de chaque côté de 10-8

Ajout de 12



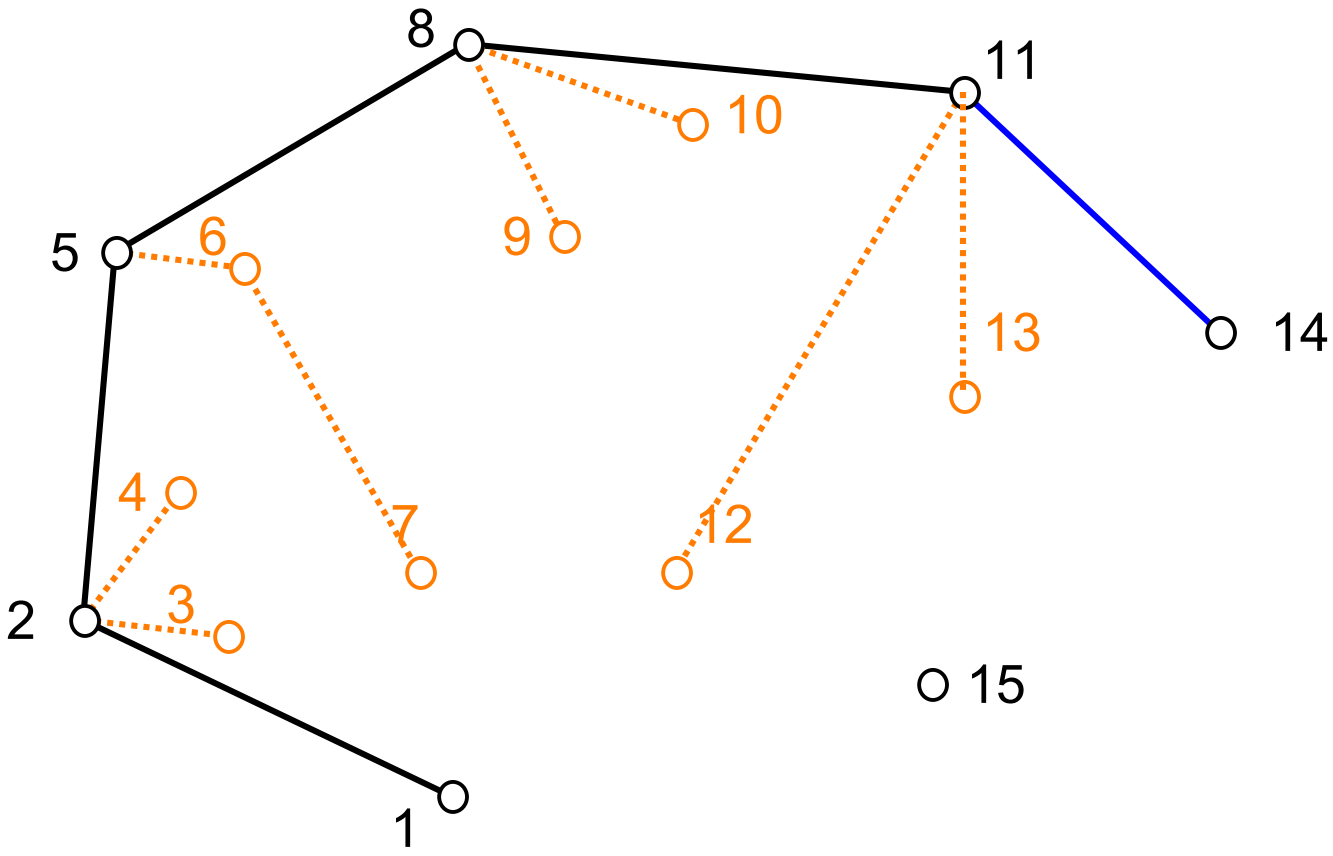
Rien à supprimer

Ajout de 13



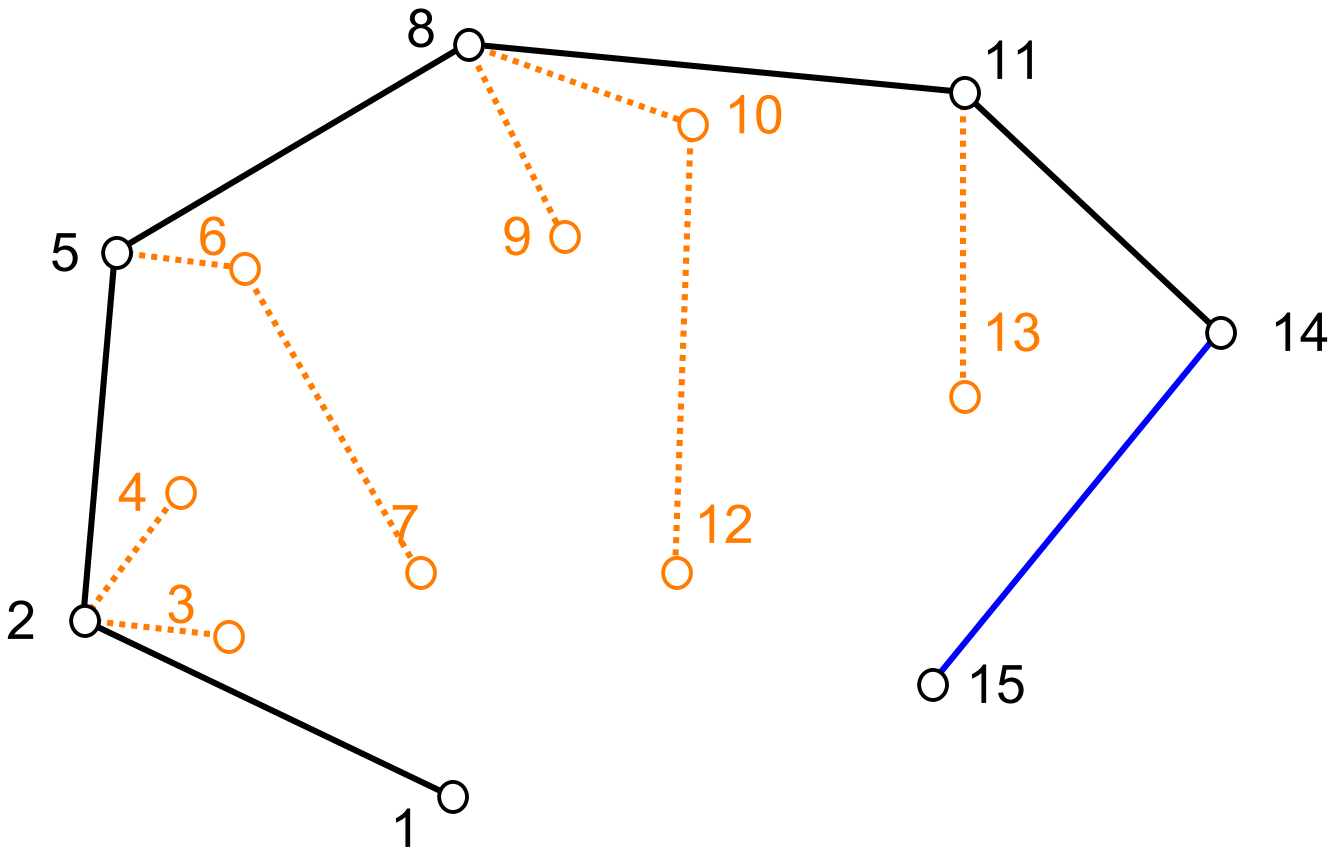
Suppression de 12 car 1 et 13 de chaque côté de 12-11

Ajout de 14



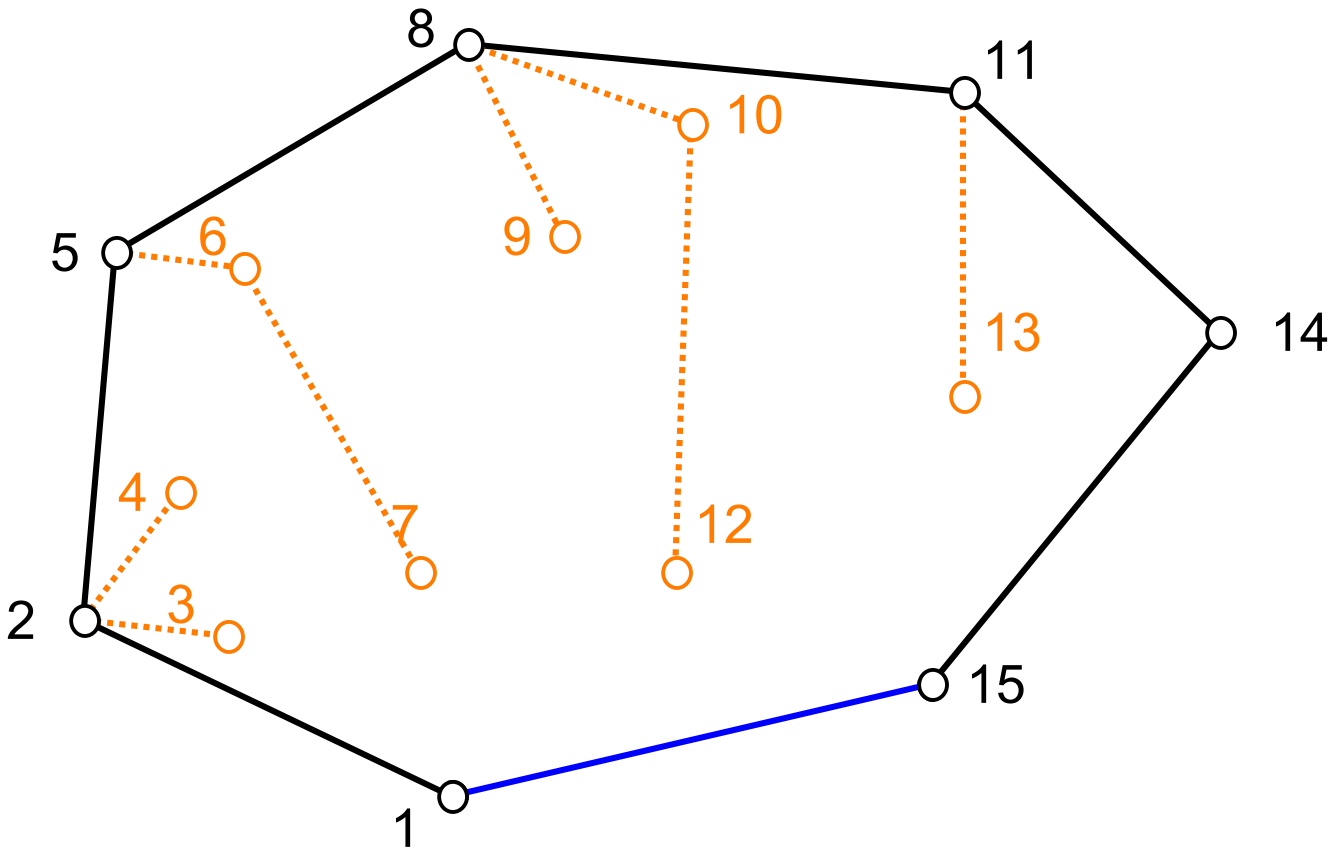
Suppression de 13 car 1 et 14 de chaque côté de 13-11

Ajout de 15

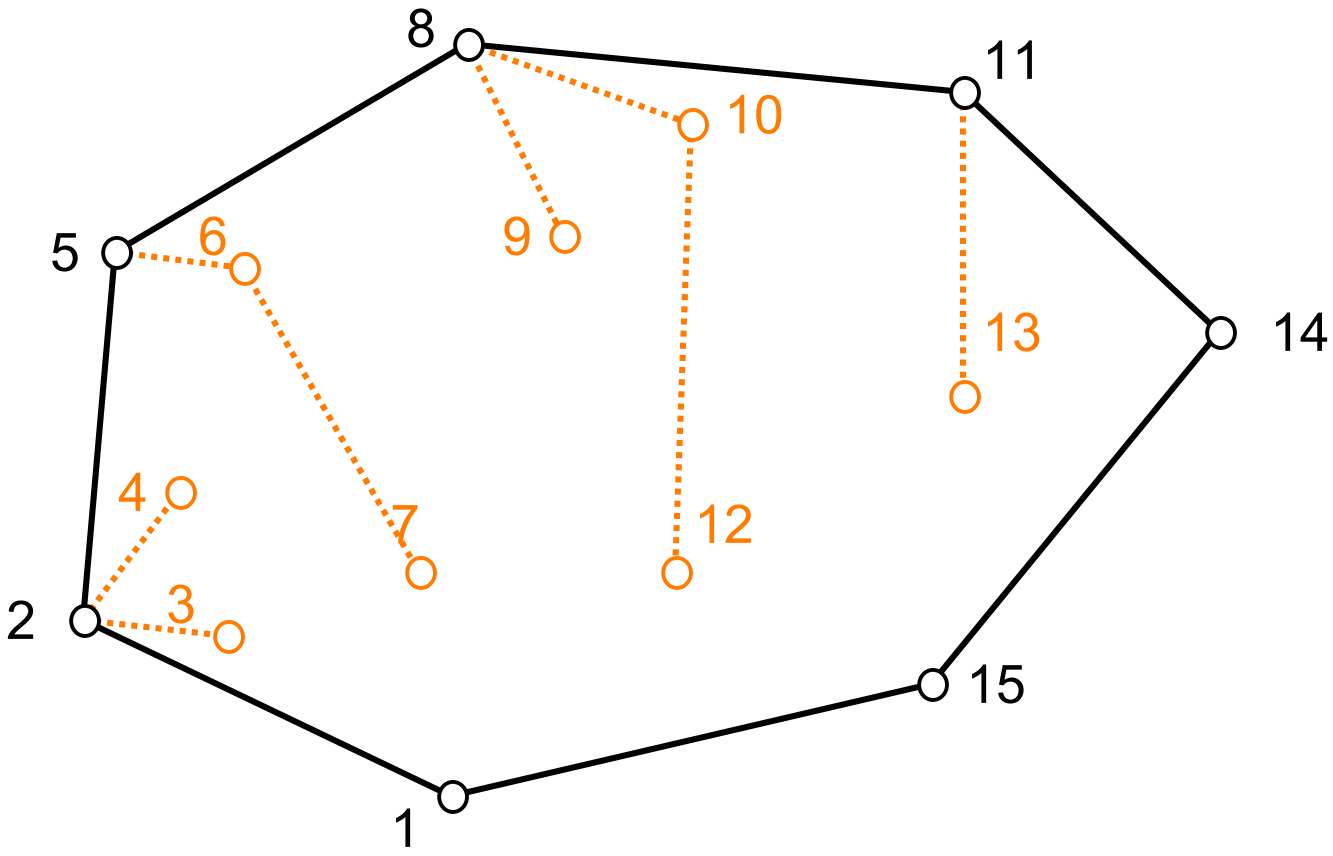


Rien à supprimer

Et c'est fini !

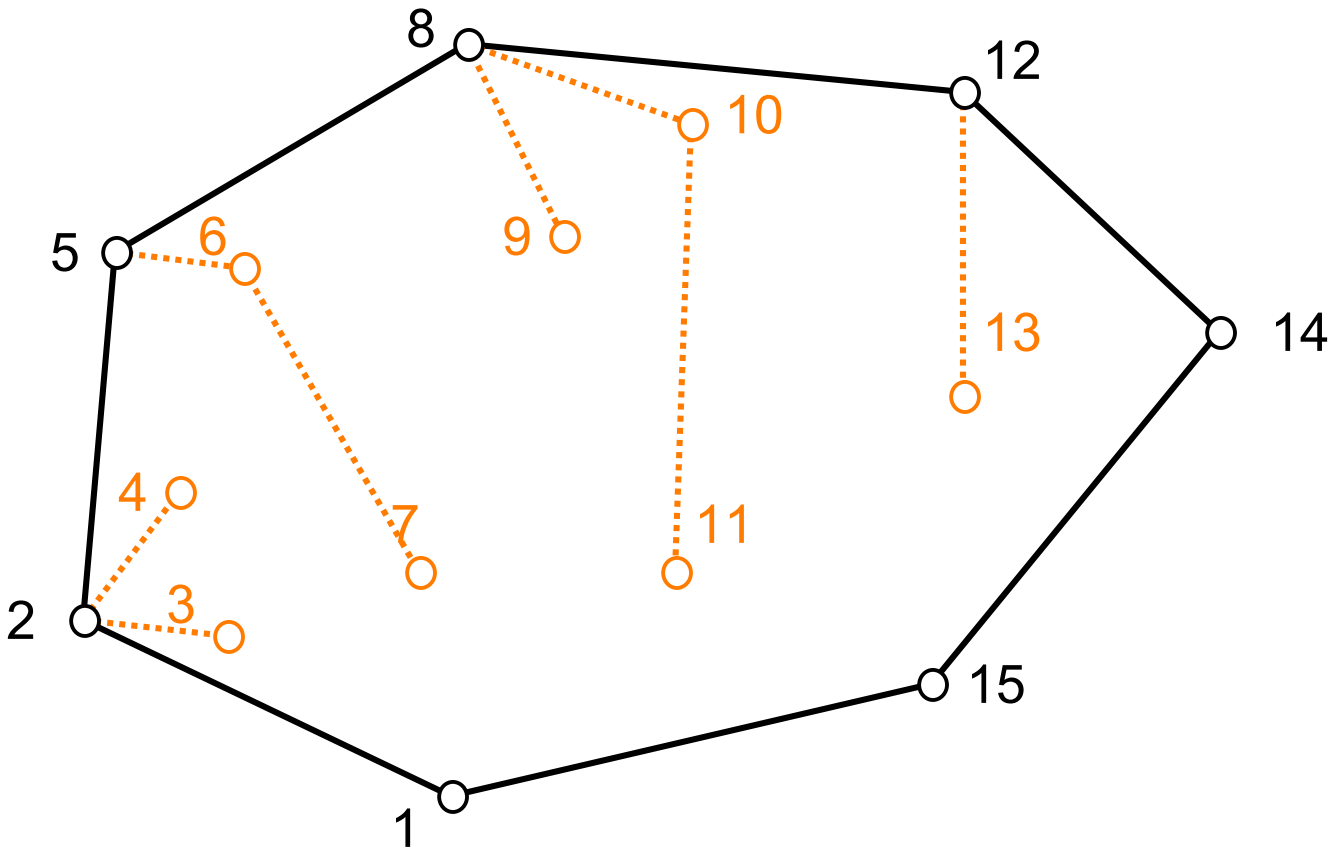


Et c'est fini !



Coût de la construction: O(n)
car chaque point est introduit et éliminé au plus une fois

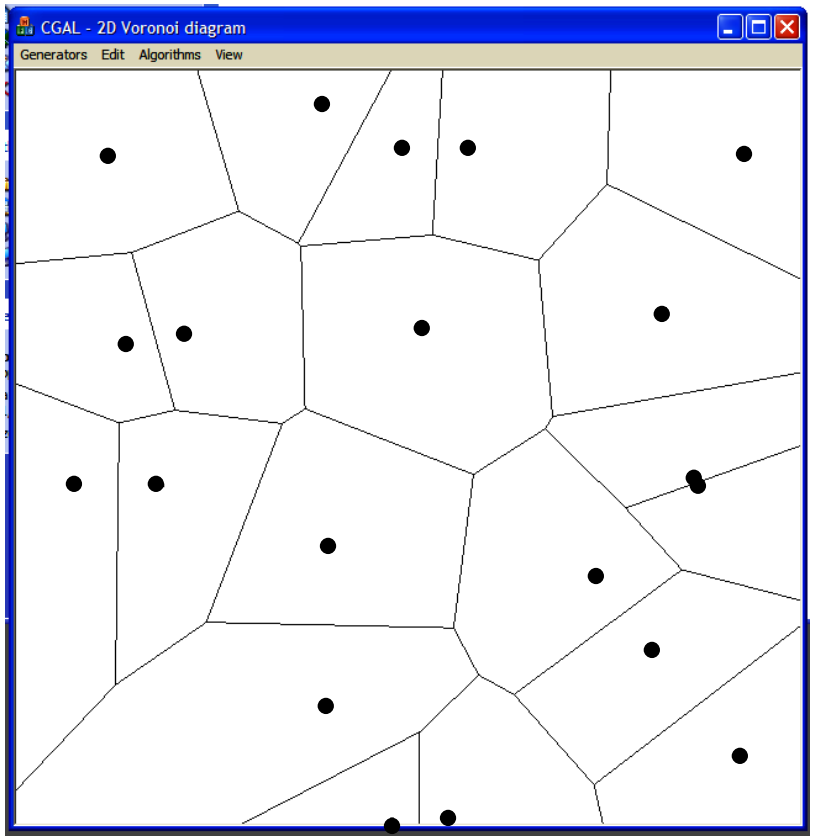
Etonnant, non ?



Coût total : O(n log(n))
L'opération chère est le tri !

Diagrammes de Voronoi

Etant donné un ensemble de points, diviser l'espace en zones de plus grande proximité à chaque point



http://www-sop.inria.fr/geometrica/team/Pierre.Alliez/demos/voronoi/

Applications

- Astronomie : Cluster d'étoiles (Descartes!)
- Ecologie : territoire des animaux, compétition des plantes
- Météo : calcul des pluies à partir de mesures ponctuelles
- Physiologie : transport de l'oxygène dans les muscles
- Métallurgie : croissance des grains dans les métaux
- Analyse numérique : recherche de bons maillages
- ...

Le problème SAT

Une formule propositionnelle est-elle toujours vraie?
peut-elle être rendue vraie?

$(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$ toujours

$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$ parfois

$(A \Rightarrow B) \Rightarrow (A \wedge \neg B))$ jamais

Le problème SAT

Une formule propositionnelle est-elle toujours vraie?
peut-elle être rendue vraie?

$(A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C))$ toujours

$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$ parfois

$(A \Rightarrow B) \Rightarrow (A \wedge \neg B))$ jamais

Applications industrielles et scientifiques

- CAO et vérification de circuits
- vérification de programmes
- études de réactions biochimiques

Mais sur beaucoup de variables !

- 50 000 ! 1000 000 ! plus?

Interprétation Booléenne

voir les connecteurs logiques comme des opérations

faux = 0 vrai = 1

0	1
1	0

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\Rightarrow$	0	1
0	1	1
1	0	1

$\Leftrightarrow$	0	1
0	1	0
1	0	1

Interprétation Booléenne

voir les connecteurs logiques comme des opérations

faux = 0 vrai = 1

0	1
1	0

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\Rightarrow$	0	1
0	1	1
1	0	1

$\Leftrightarrow$	0	1
0	1	0
1	0	1

méthode bête: essayer toutes les valeurs de vérité

A=0 B=0 C=1 D=1 : $(A \Leftrightarrow B) \wedge (C \Leftrightarrow D) = 1$

A=0 B=1 C=0 D=1 : $(A \Leftrightarrow B) \wedge (C \Leftrightarrow D) = 0$

Interprétation Booléenne

voir les connecteurs logiques comme des opérations

faux = 0 vrai = 1

0	1
1	0

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\Rightarrow$	0	1
0	1	1
1	0	1

$\Leftrightarrow$	0	1
0	1	0
1	0	1

méthode bête: essayer toutes les valeurs de vérité

A=0 B=0 C=1 D=1 : $(A \Leftrightarrow B) \wedge (C \Leftrightarrow D) = 1$

A=0 B=1 C=0 D=1 : $(A \Leftrightarrow B) \wedge (C \Leftrightarrow D) = 0$

mais exponentiel dans le nombre de variables !

Un problème NP-complet !

- Il est facile (polynomial) de vérifier qu'une solution proposée en est bien une
- Mais il est difficile de trouver la bonne parmi les choix en nombre exponentiel

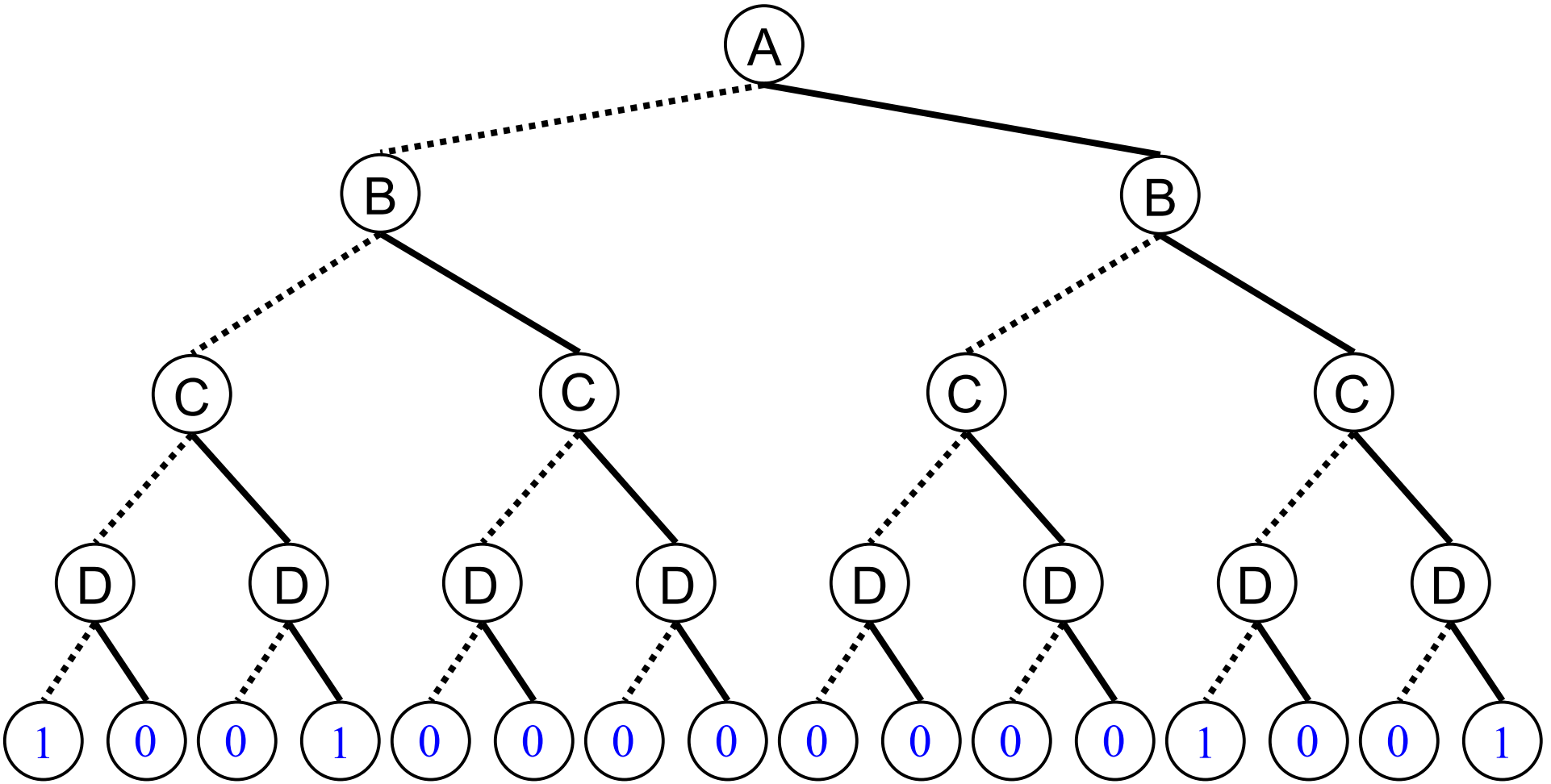
Emploi du temps, horaire de train, routage de circuits,...

La course en SAT

- BDD : forme canonique
  - donnent toutes les solutions
  - mais explosent vite en mémoire (ex: multiplication)
  - limite : quelques centaines / milliers de variables....
- Nouvelles méthodes
  - ATPG, DPLL apprentissage, Stålmark, etc.
  - cherchent une solution
  - très efficaces en mémoire, peu prédictibles en temps
- Problème ouvert: géométrie des formules faciles?
  - pratique actuelle: lancer plusieurs prouveurs en parallèle

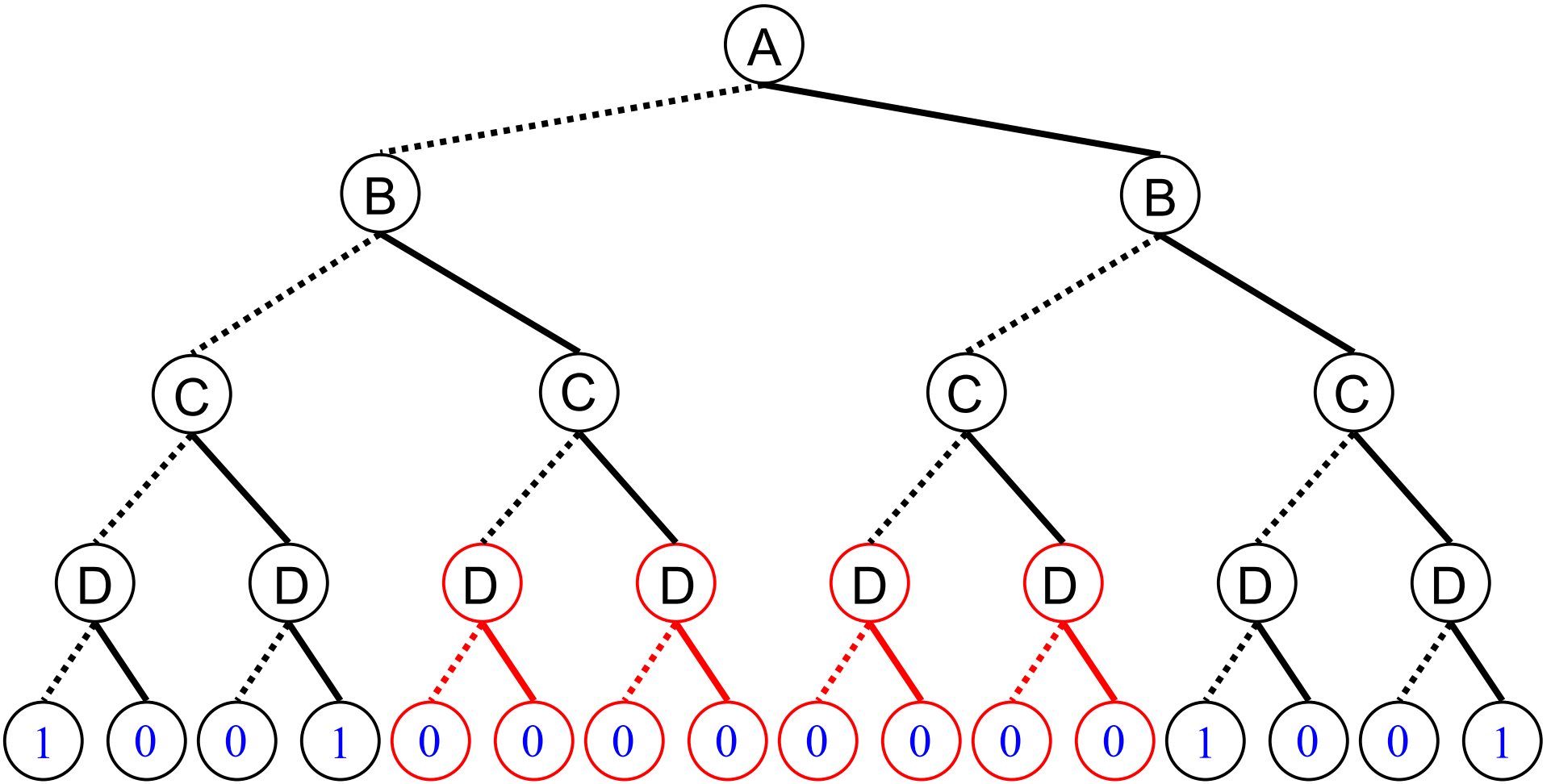
Arbre de Shannon

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



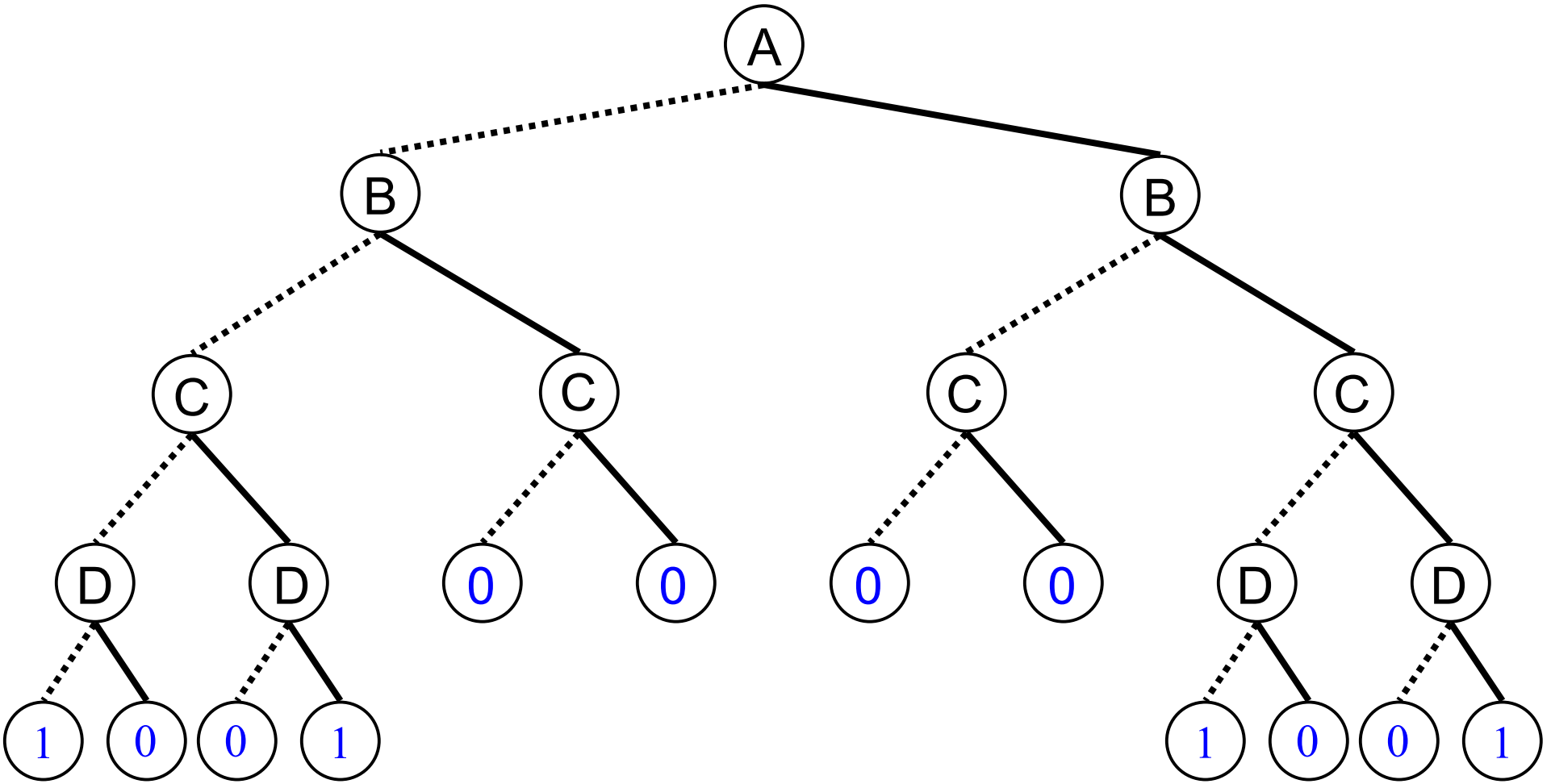
Suppression de feuilles identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



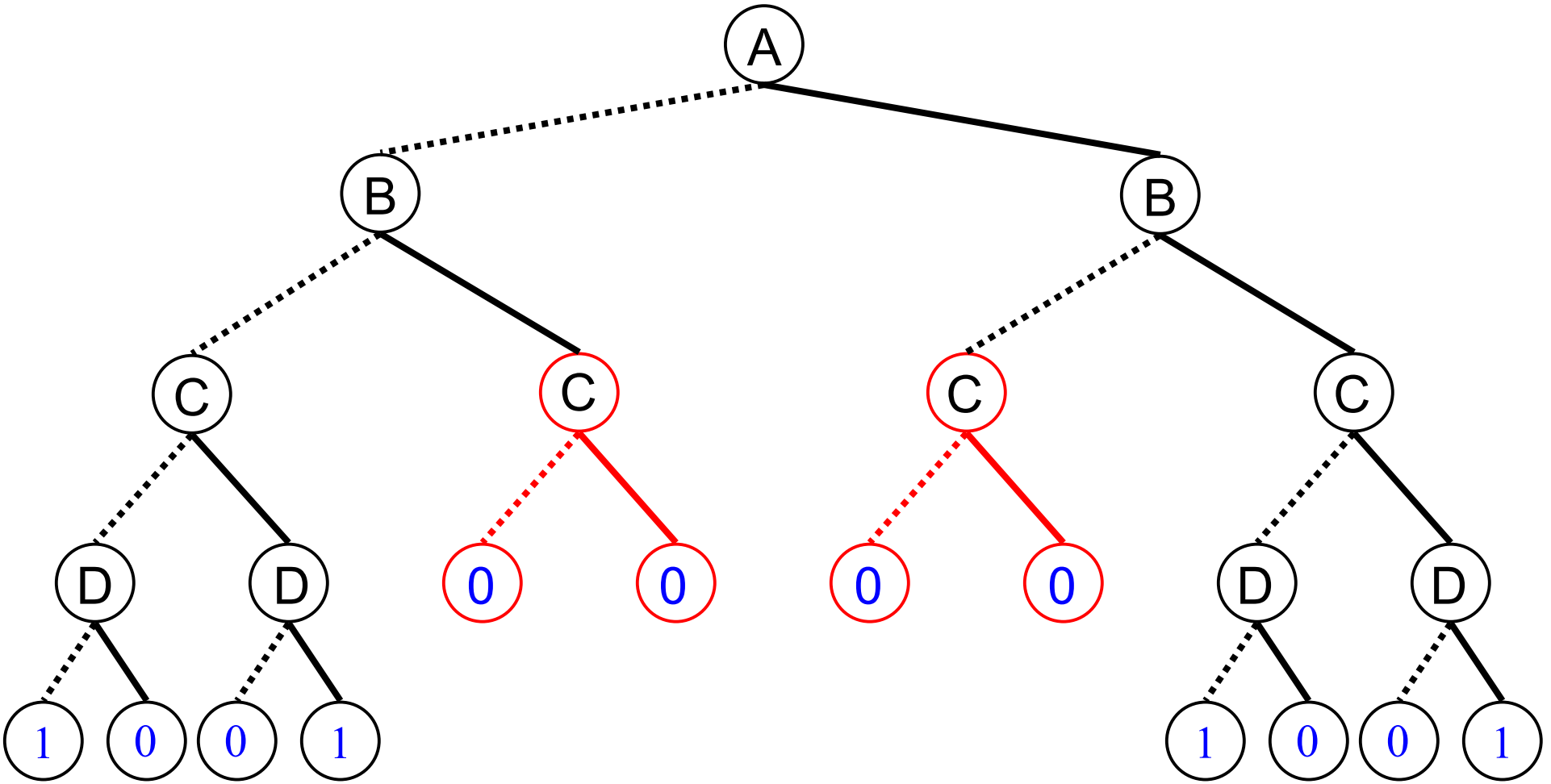
Suppression de feuilles identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



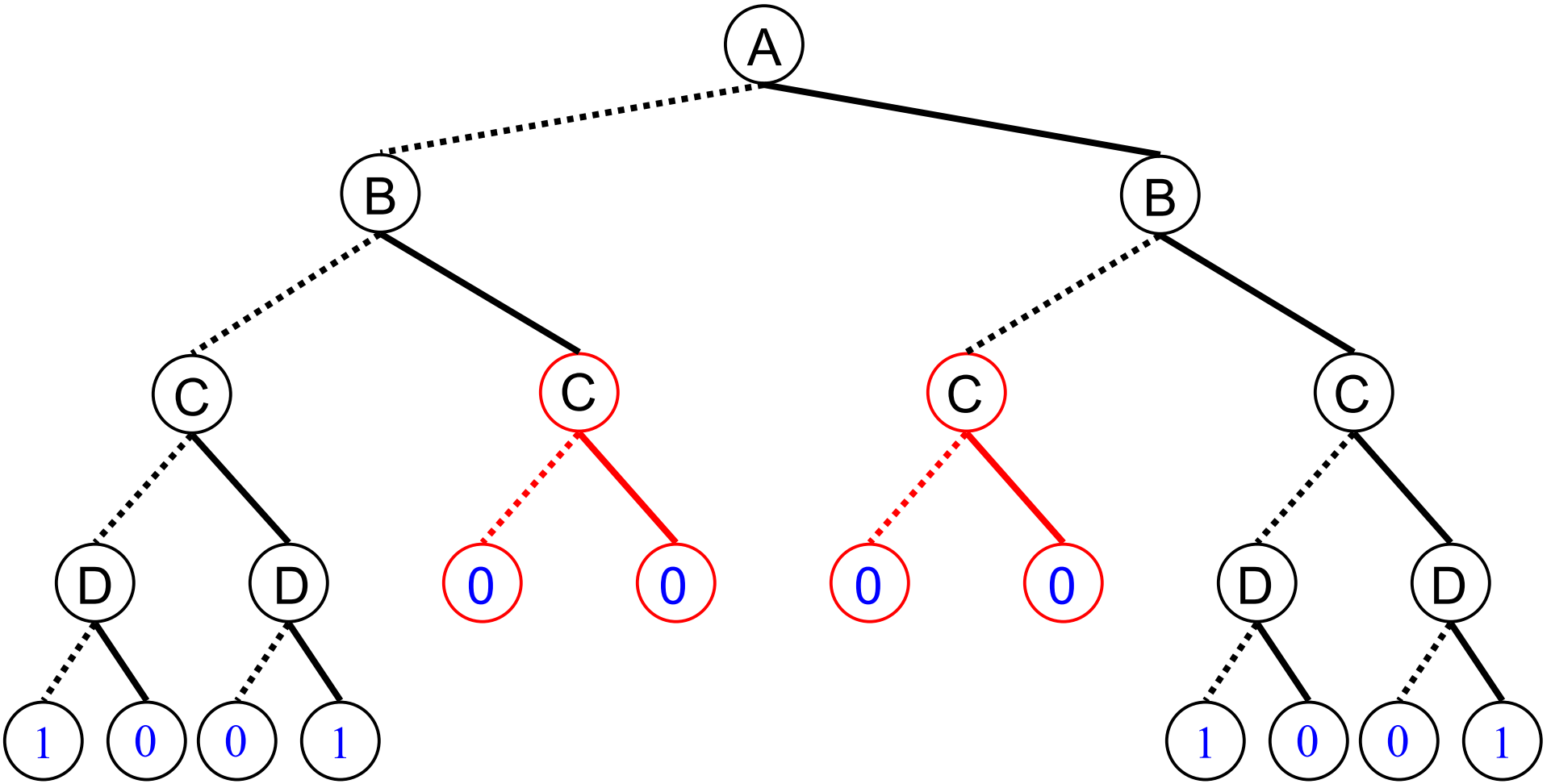
Suppression de feuilles identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



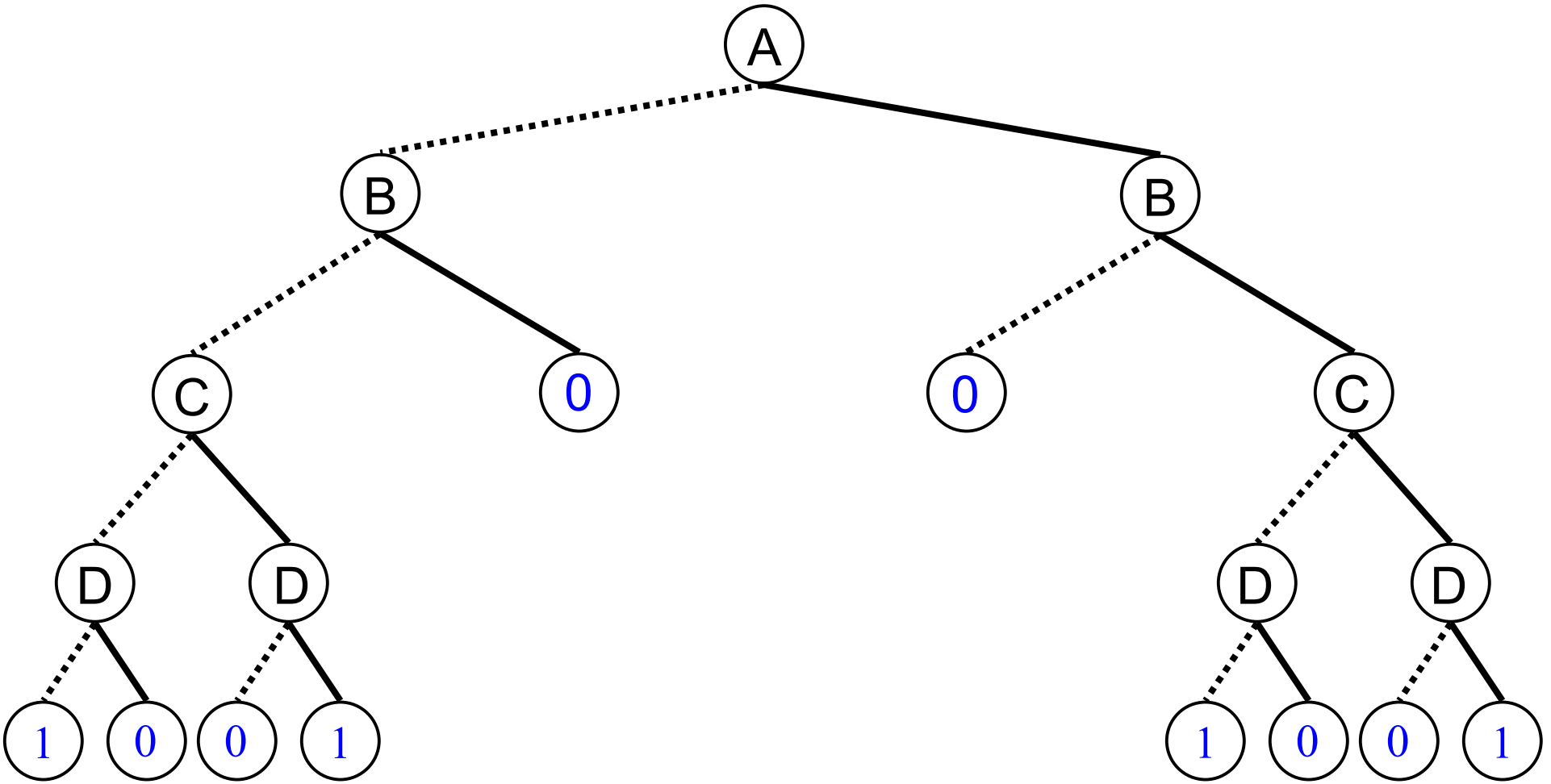
Suppression de feuilles identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



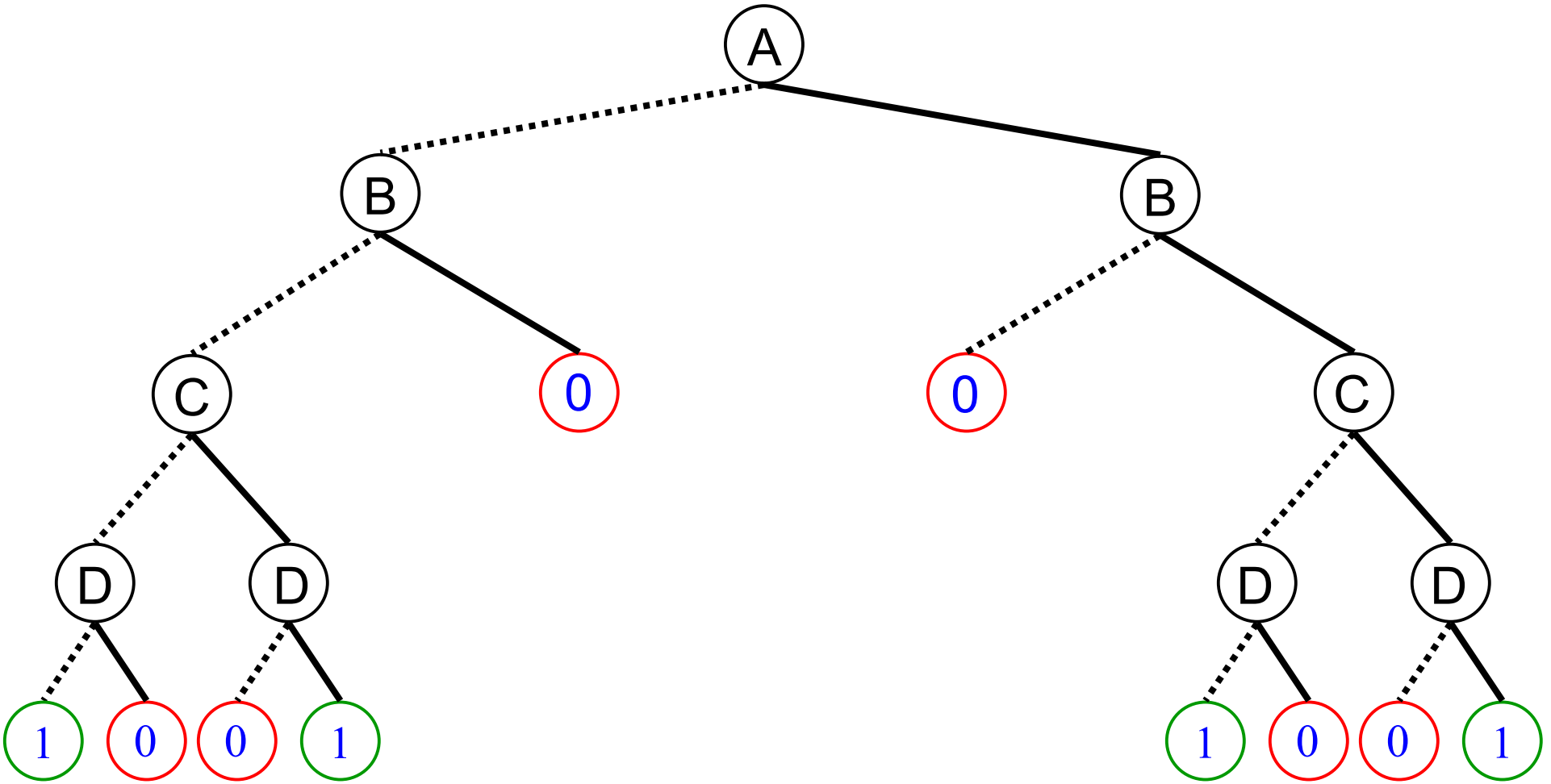
Suppression de feuilles identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



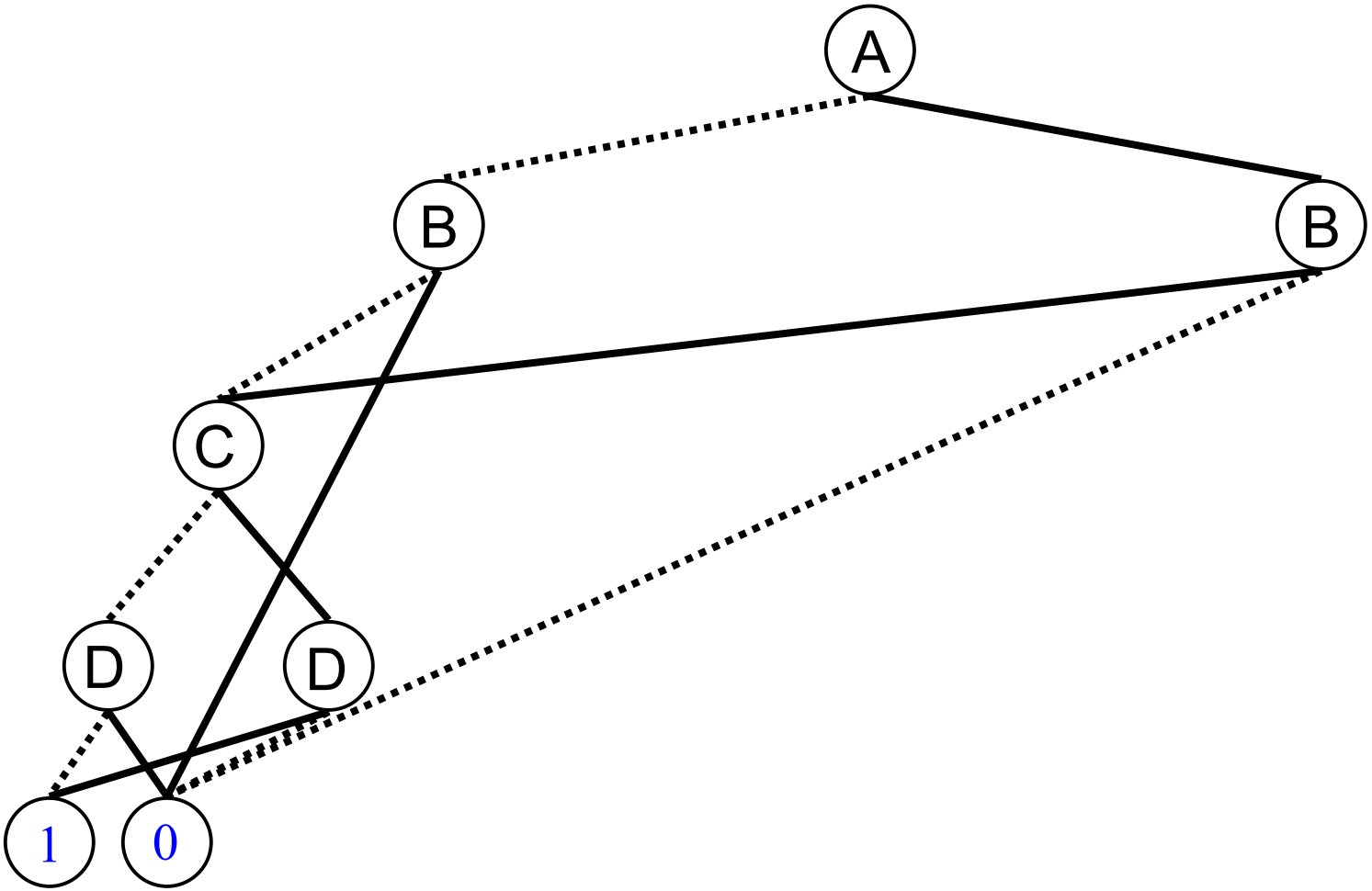
Partages de sous-arbres identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



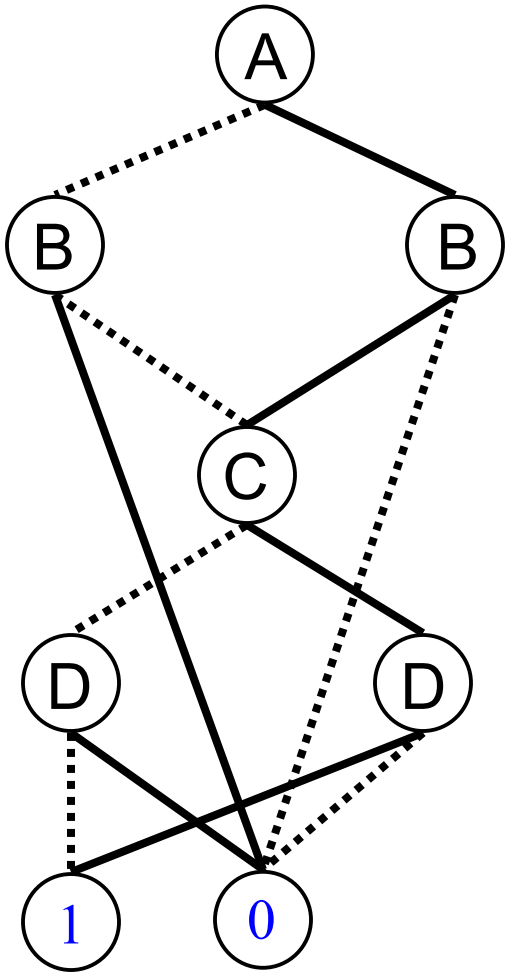
Partages de sous-arbres identiques

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$



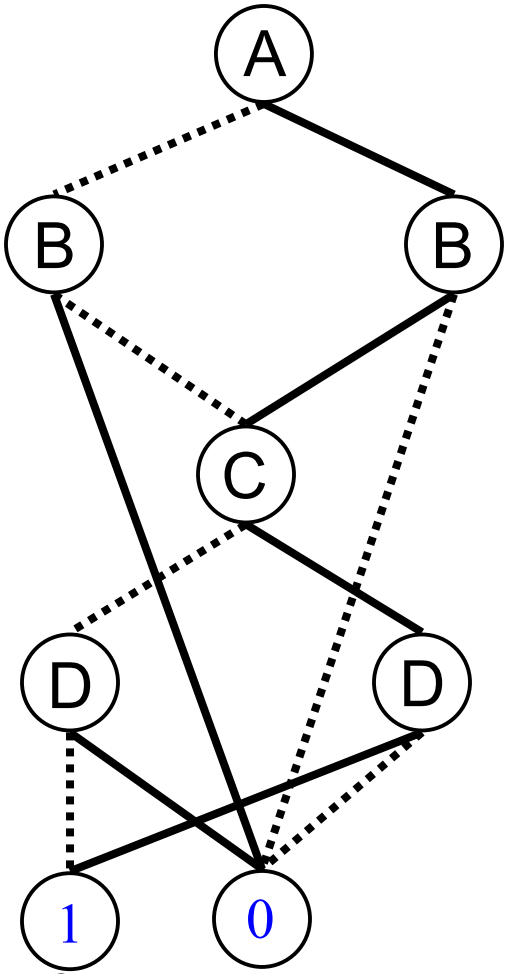
BDD = Binary Decision Diagram

$$(A \Leftrightarrow B) \wedge (C \Leftrightarrow D)$$

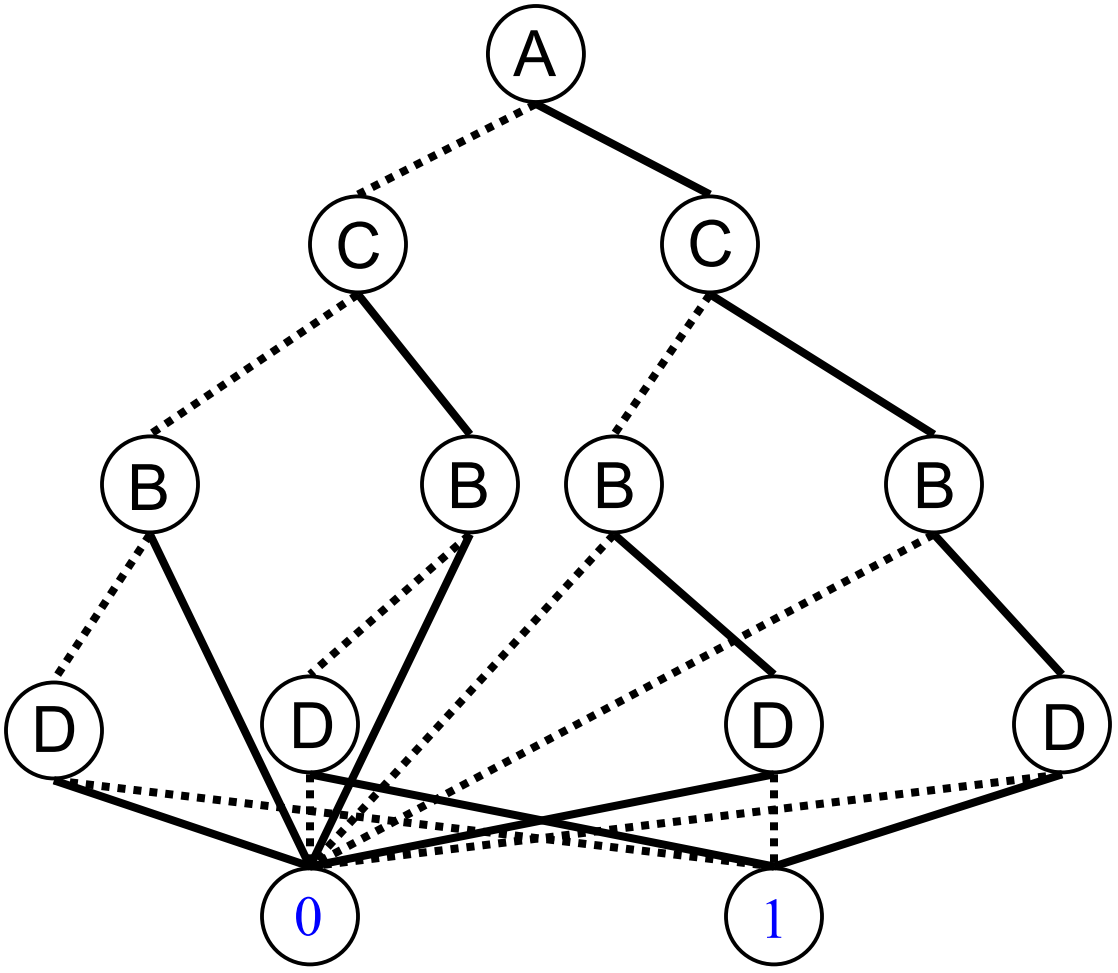


Sensibilité à l'ordre des variables

A - B - C - D



A - C - B - D



Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	~~4~~	5	~~6~~	7	~~8~~	9	~~10~~	11
~~12~~	13	~~14~~	15	~~16~~	17	~~18~~	19	~~20~~	21
~~22~~	23	~~24~~	25	~~26~~	27	~~28~~	29	~~30~~	31
~~32~~	33	~~34~~	35	~~36~~	37	~~38~~	39	~~40~~	41
~~42~~	43	~~44~~	45	~~46~~	47	~~48~~	49	~~50~~	51
~~52~~	53	~~54~~	55	~~56~~	57	~~58~~	59	~~60~~	61
~~62~~	63	~~64~~	65	~~66~~	67	~~68~~	69	~~70~~	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	~~25~~	26	27	28	29	30	31
32	33	34	~~35~~	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	~~55~~	56	57	58	59	60	61
62	63	64	~~65~~	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	~~49~~	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

Le crible d'Eratosthène

- Un nombre est premier s'il n'a pas d'autre diviseur que 1 et lui-même

2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21
22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41
42	43	44	45	46	47	48	49	50	51
52	53	54	55	56	57	58	59	60	61
62	63	64	65	66	67	68	69	70	71

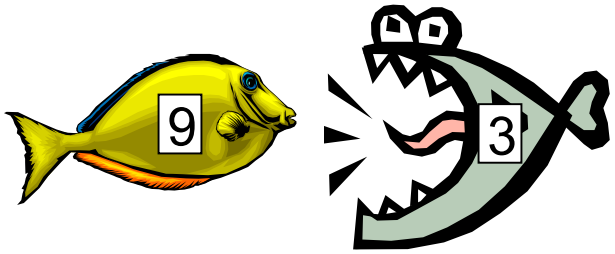
Eratosthène-Darwin : p, kp → p

Eratosthène-Darwin :

p, kp → p

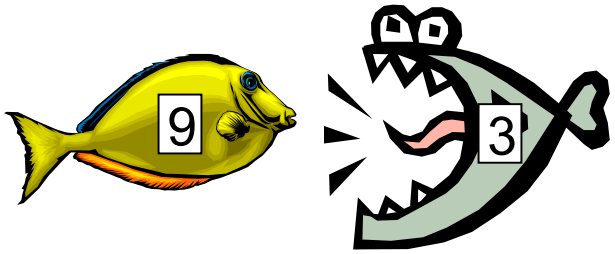
Eratosthène-Darwin :

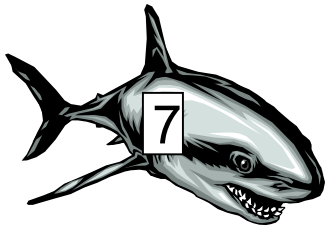
p, kp → p

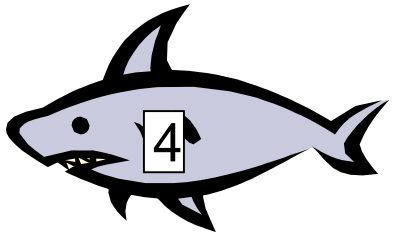


Eratosthène-Darwin :

p, kp → p

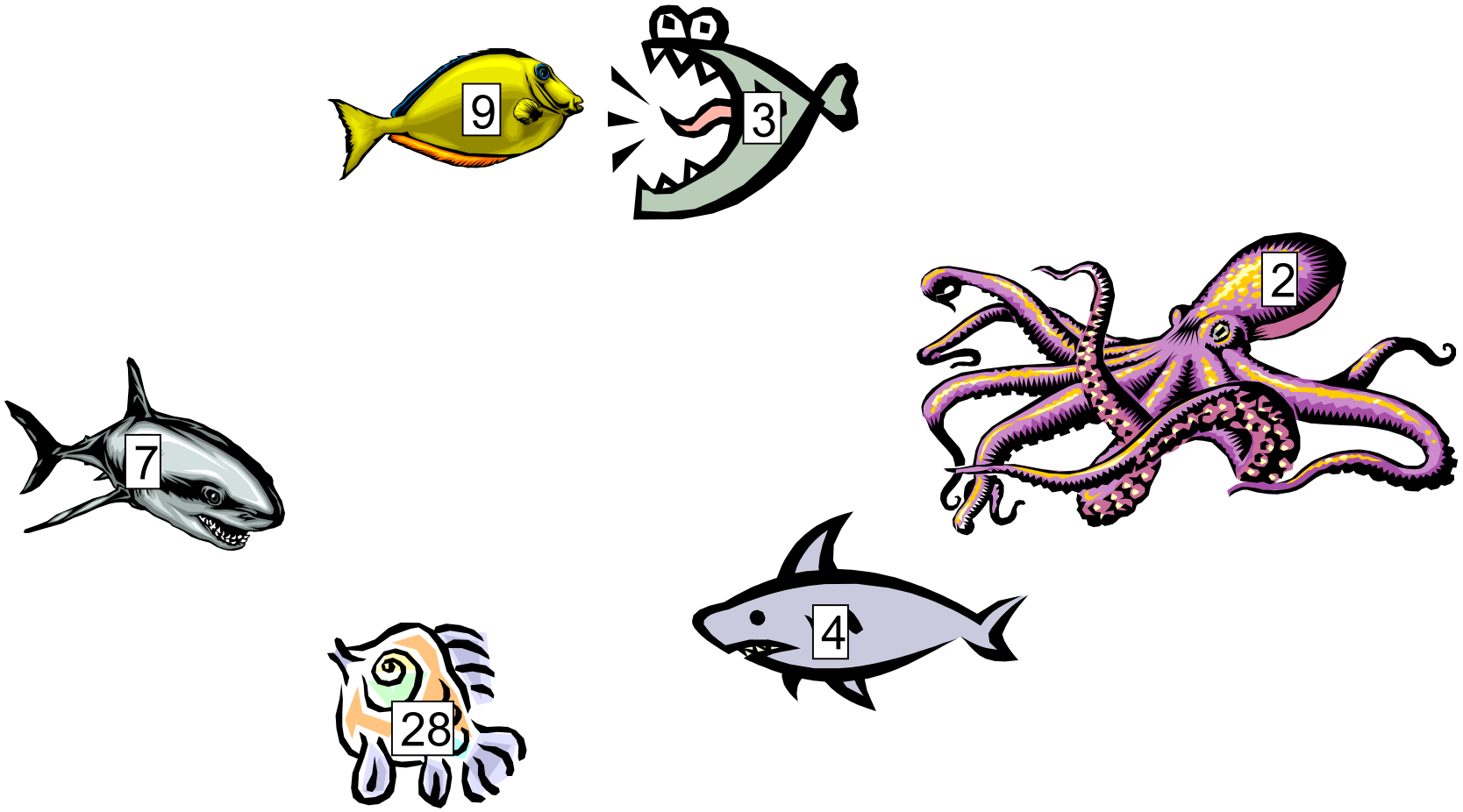






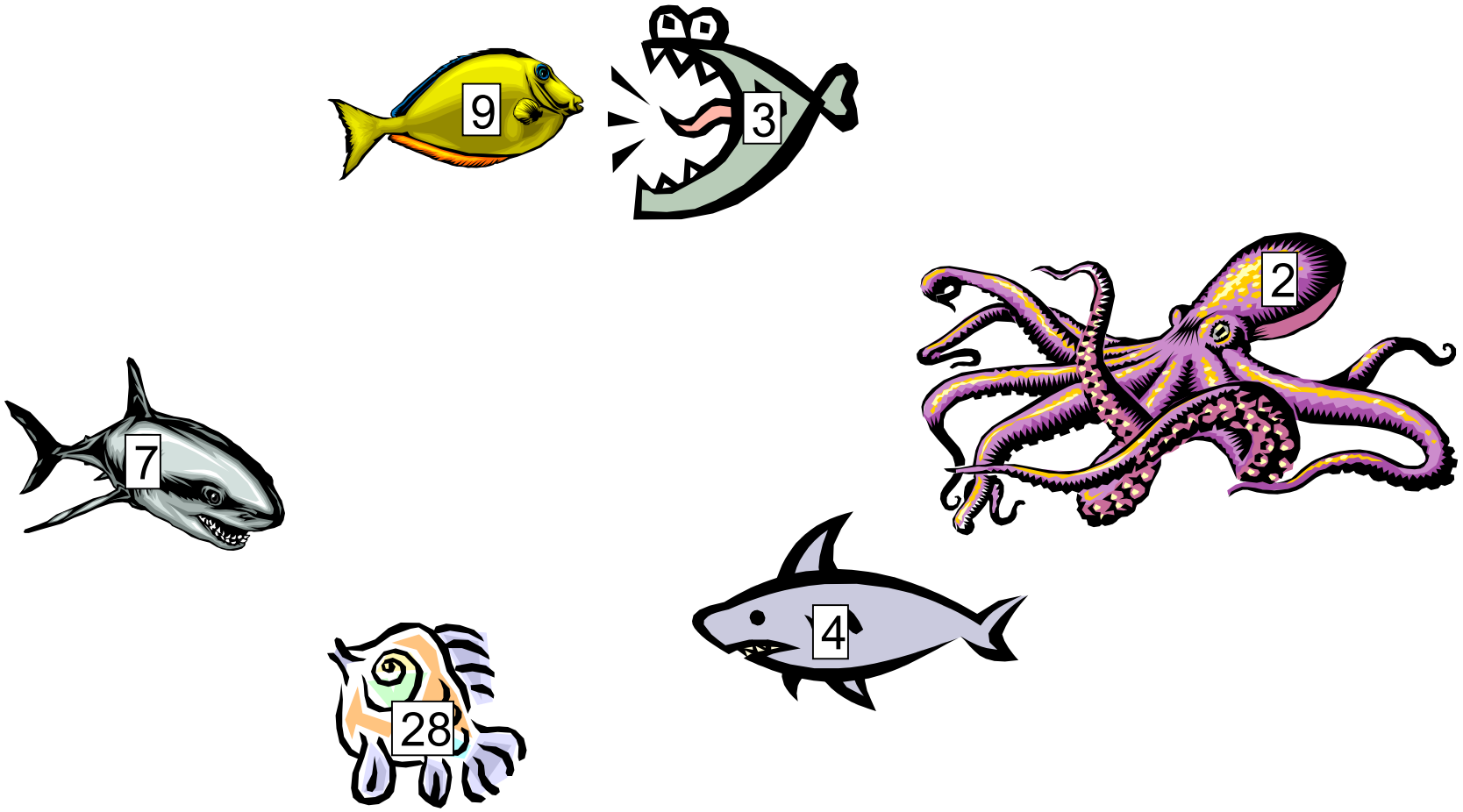
Eratosthène-Darwin :

p, kp → p



Eratosthène-Darwin :

p, kp → p



Banâtre - Le Métayer (INRIA)

Algorithmes distribués

- Protocoles de communication
- gestion de mémoire partagée multiprocesseurs
- gestion de cache / pipeline de microprocesseurs
- bases de données réparties
- gestion de trafic aérien ou automobile
- ...

Cf. cours circuits, systèmes embarqué, réseaux,...

Les nombres négatifs, Brahmagupta, 628

Une dette moins zéro est une dette

Un bien moins zéro est un bien

Zéro moins zéro est nul

Une dette retranchée de zéro est un bien

Alors qu'un bien retranché de zéro est une dette

Le produit de zéro par une dette ou un bien est zéro

le produit ou le quotient d'une dette par un bien est une dette

le produit ou le quotient de deux dettes est un bien

....

Quid de l'Europe?

D'un professeur à un parent, XV^{e} siècle

Si vous voulez vous contenter de ne lui faire apprendre que la pratique des additions et des soustractions, alors n'importe quelle université allemande ou française fera l'affaire. Par contre, si vous tenez à pousser son instruction jusqu'à la multiplication ou à la division, si tant qu'il en soit capable, alors il vous faudra l'envoyer dans les écoles italiennes.