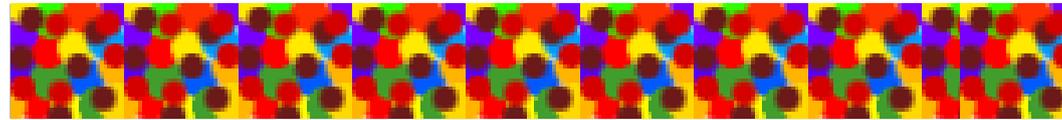
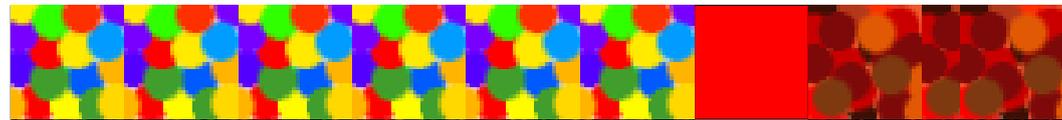


Arbres binaires de recherche

Idée qui vient du tri rapide



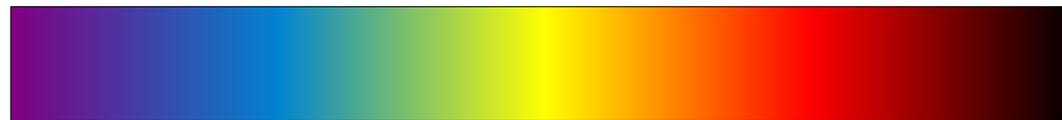
diviser

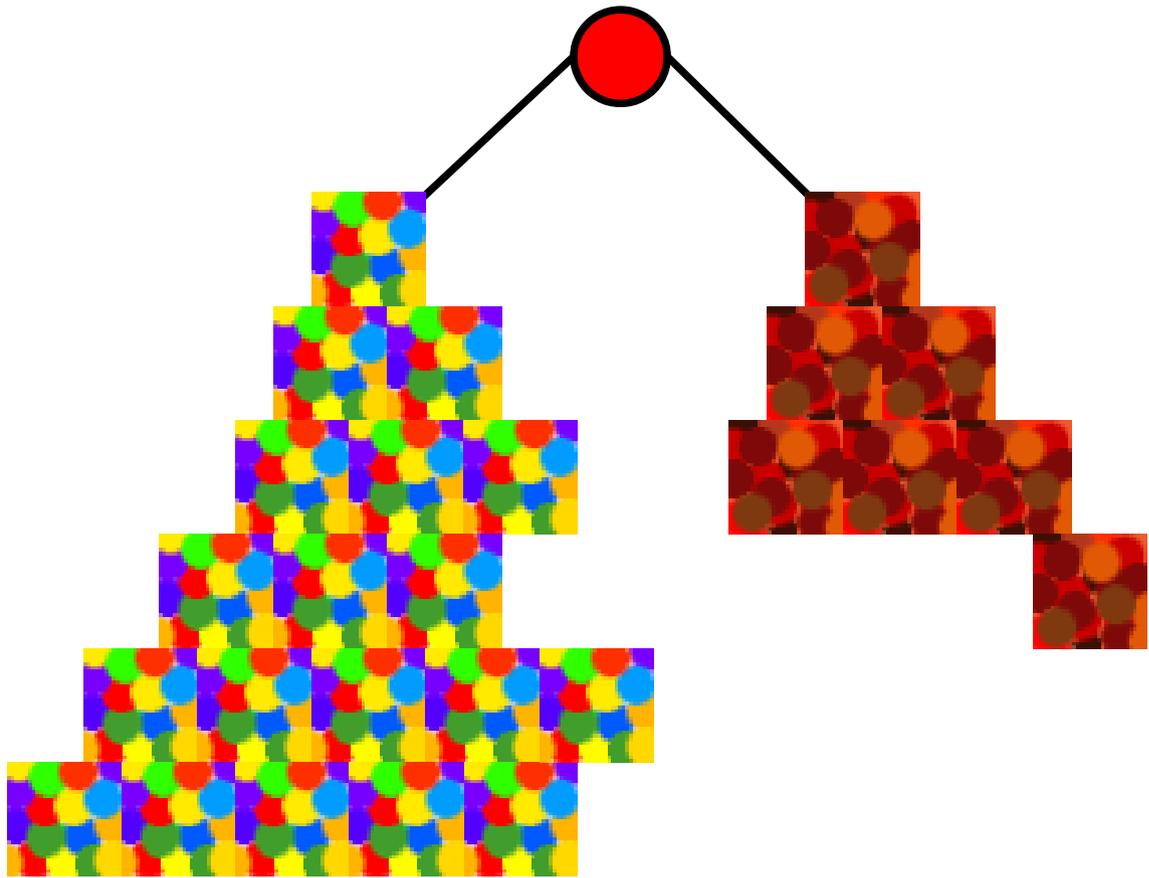
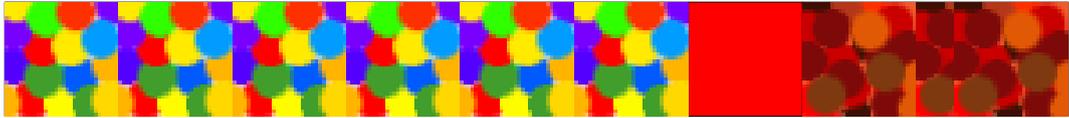


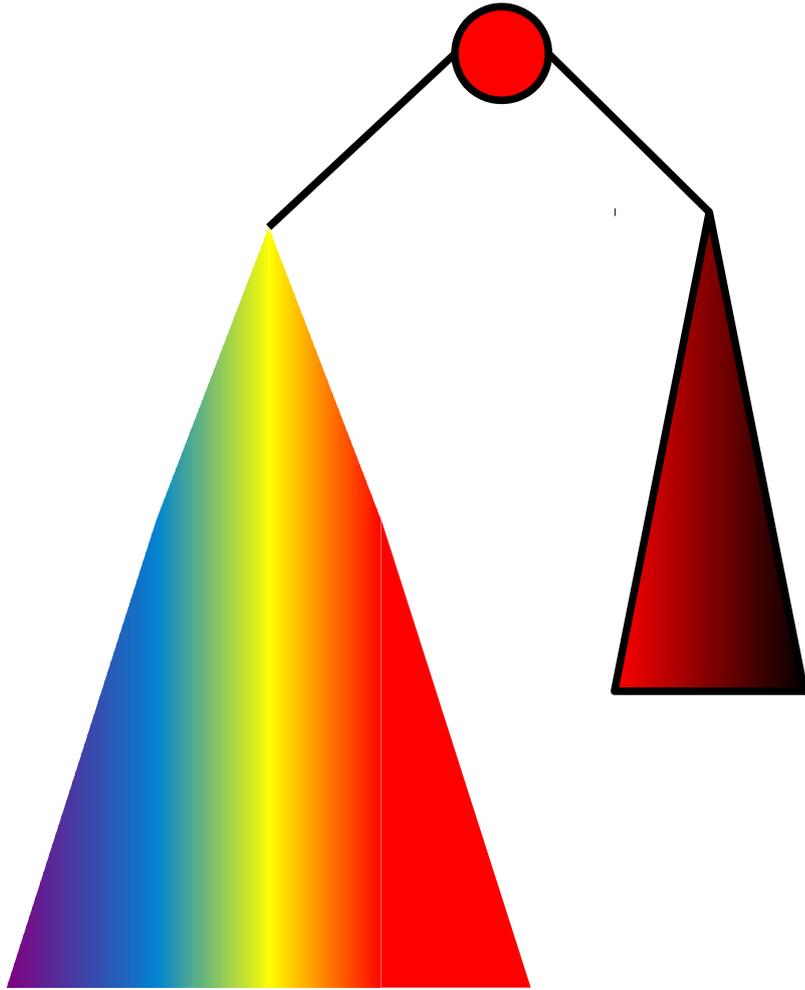
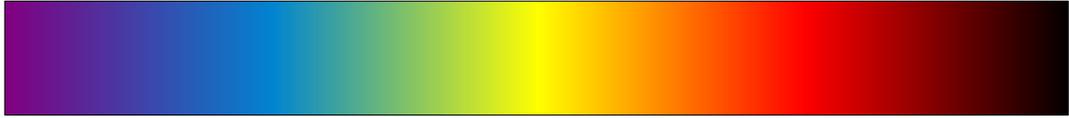
régner



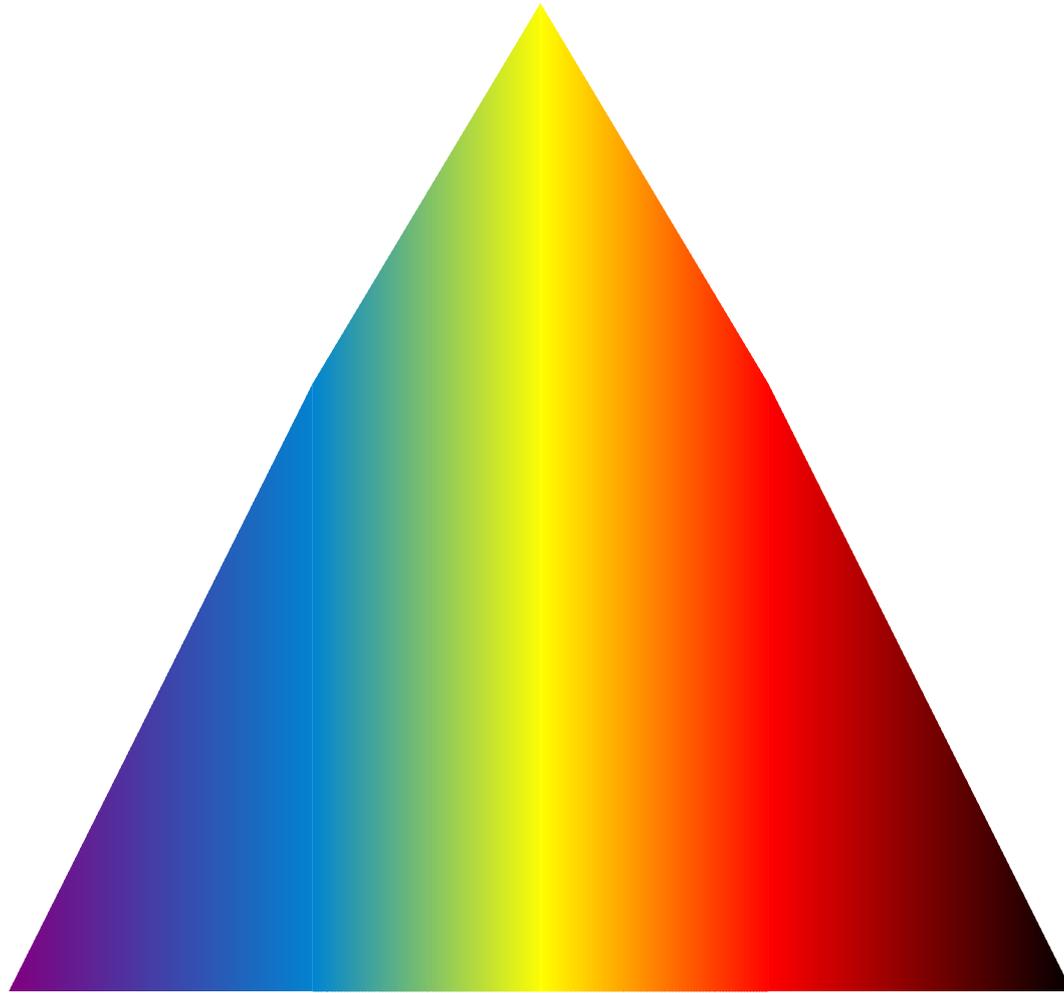
régner



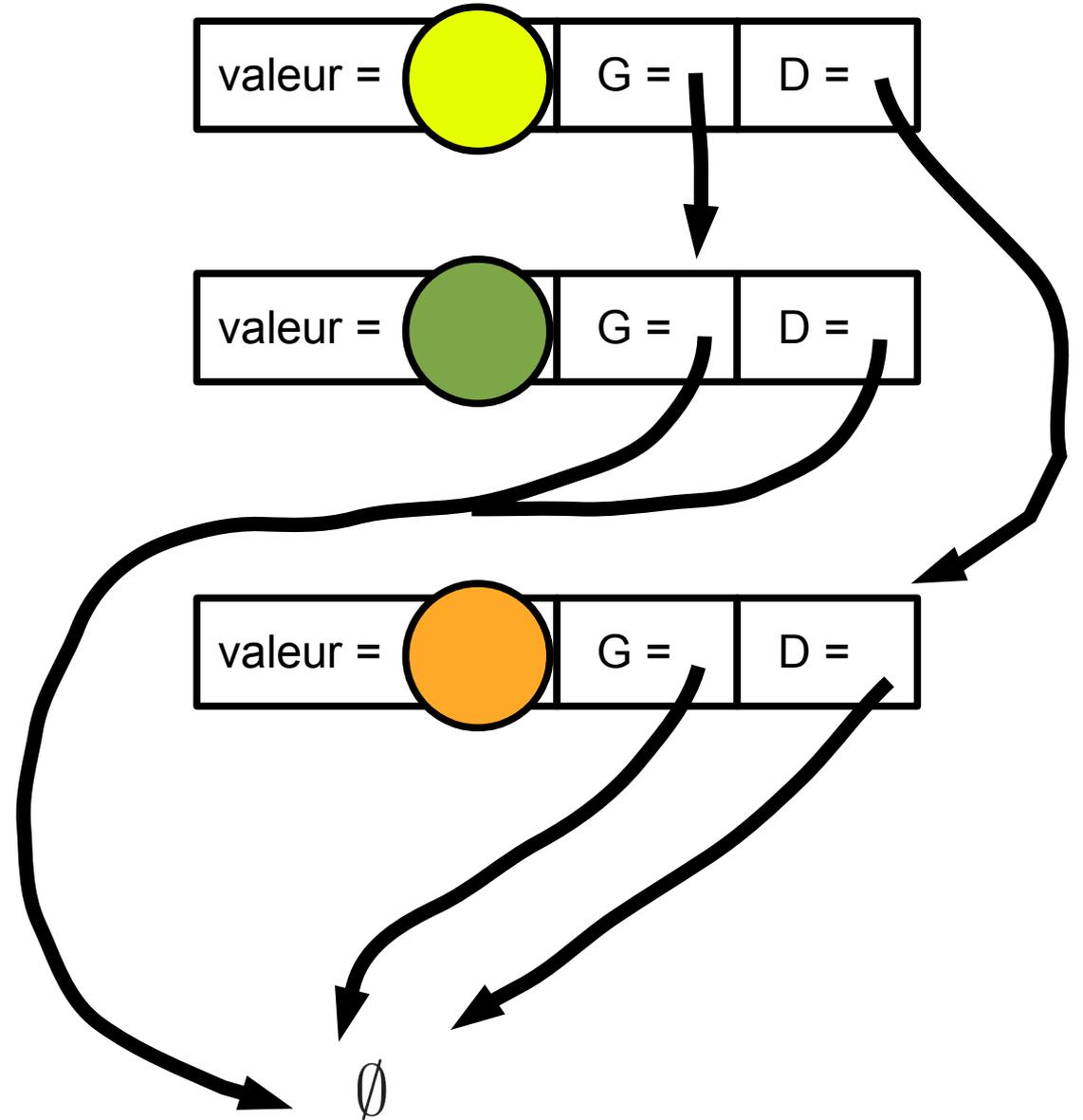
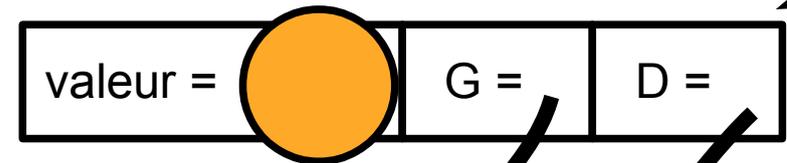
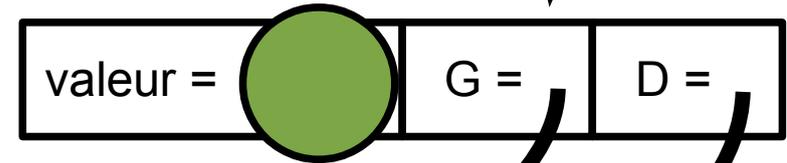
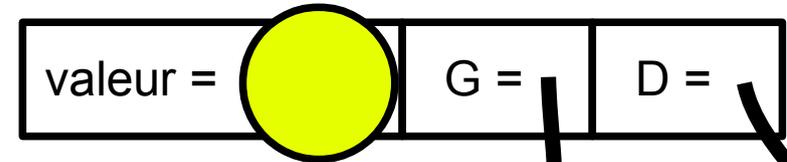
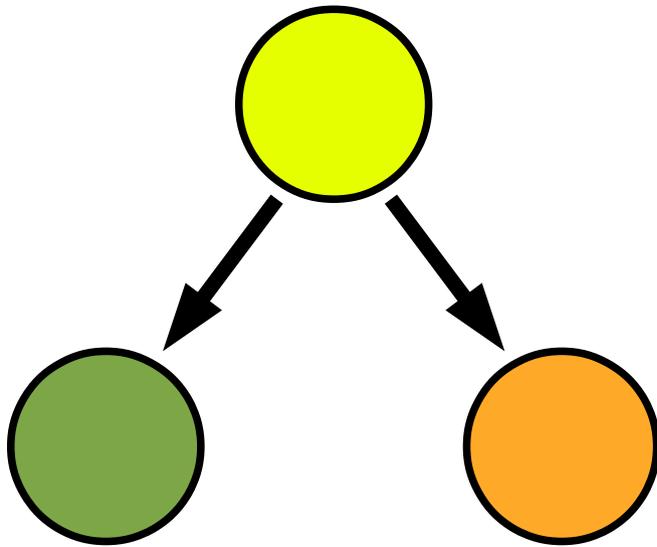




Un arbre de recherche



Représentation en machine

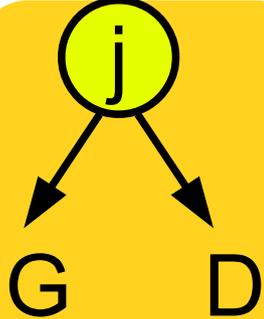


Etude de cas

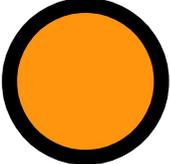
Etude de cas(T)

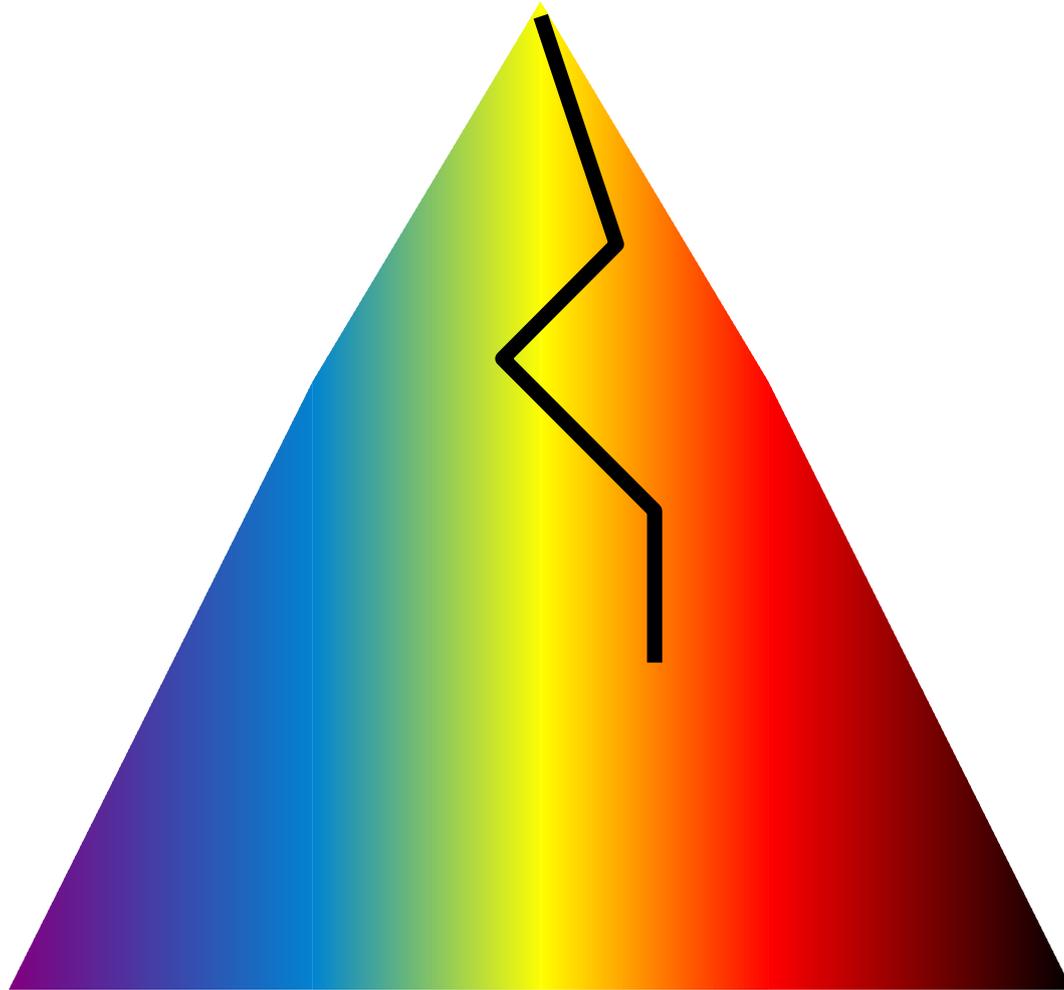
\emptyset

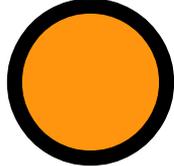
...

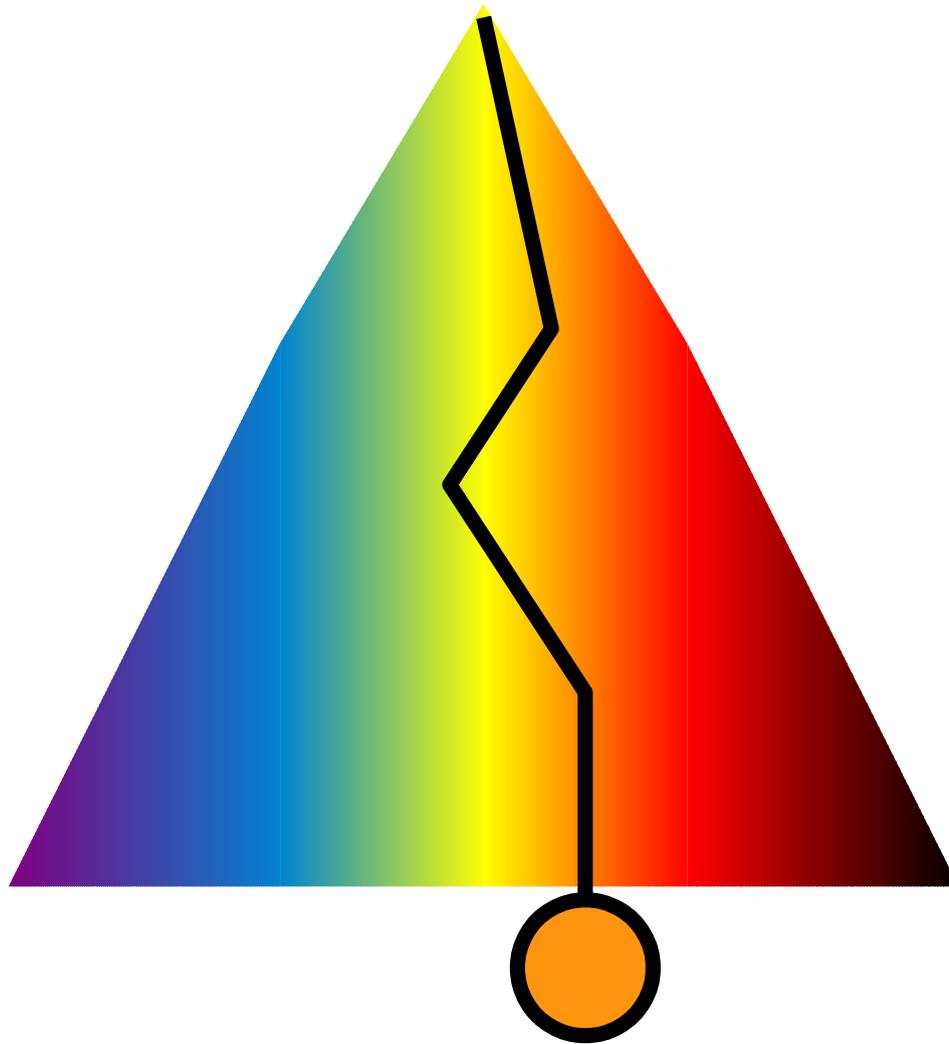


...

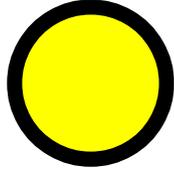
Rechercher 

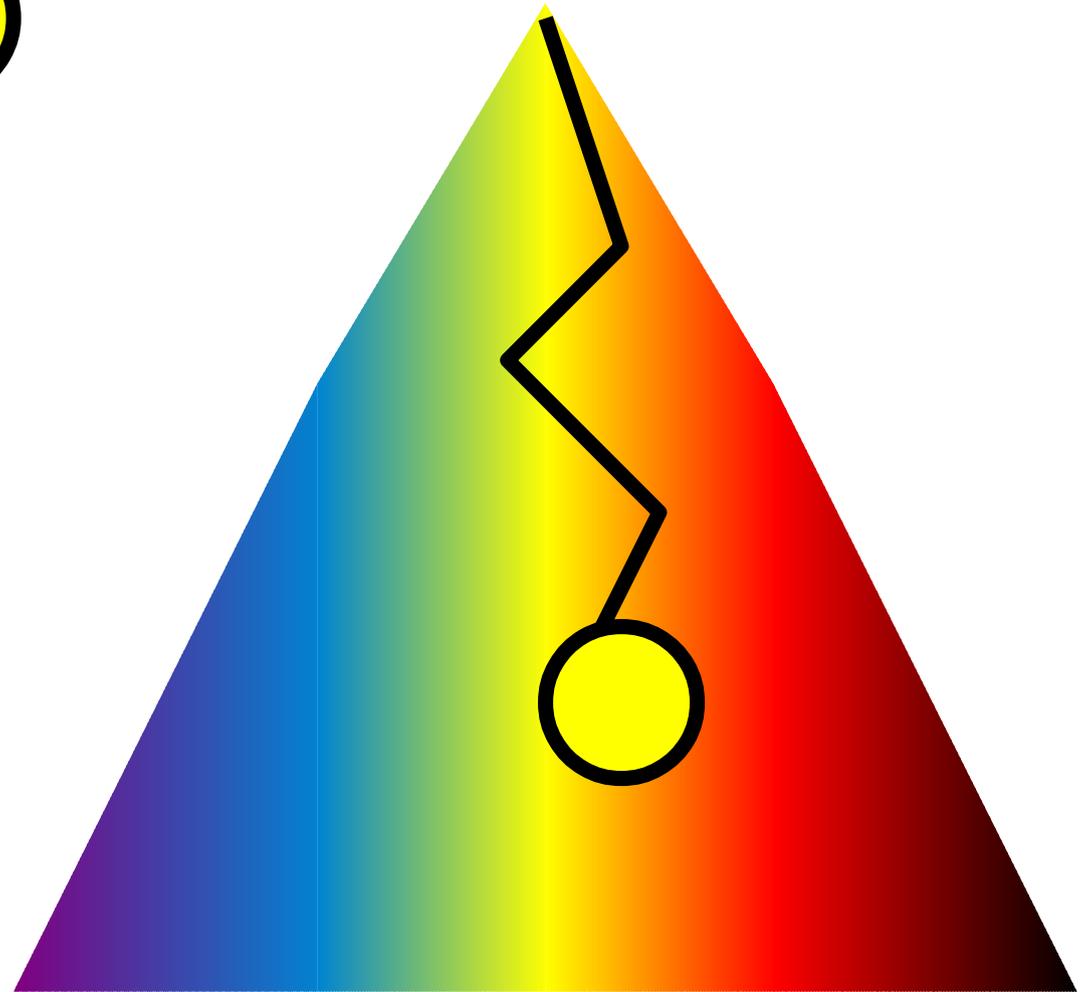


Ajouter 



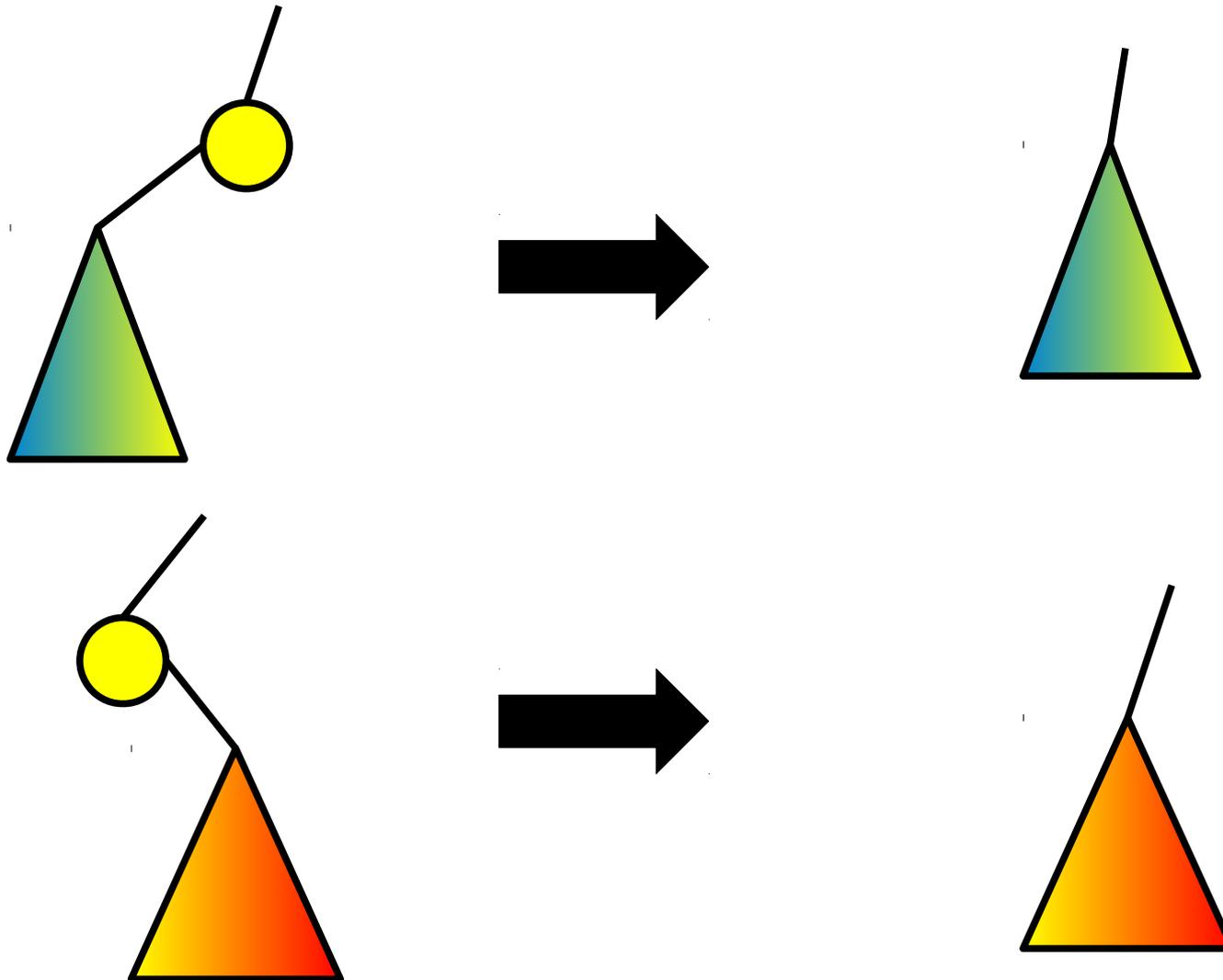
Supprimer

1) Recherchons 



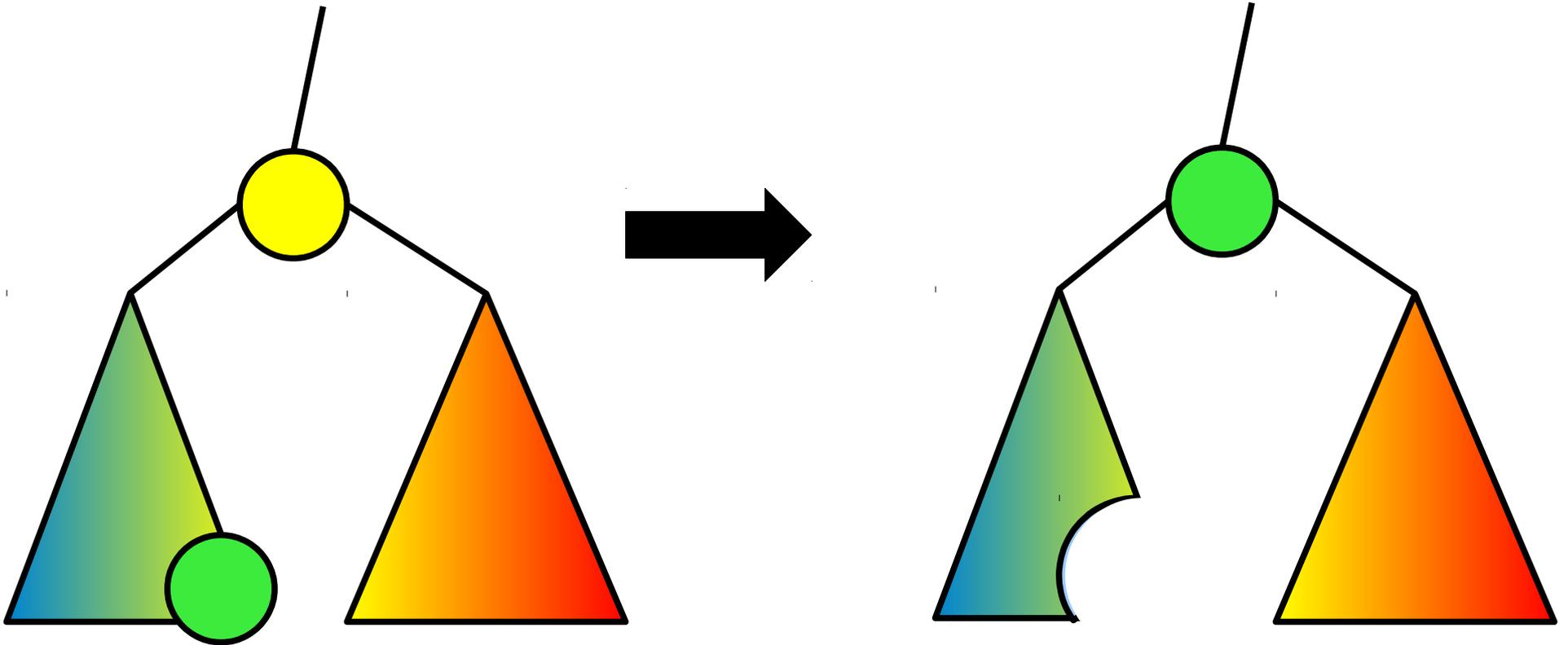
Supprimer

2) Supprimons effectivement



Supprimer

2) Supprimons effectivement



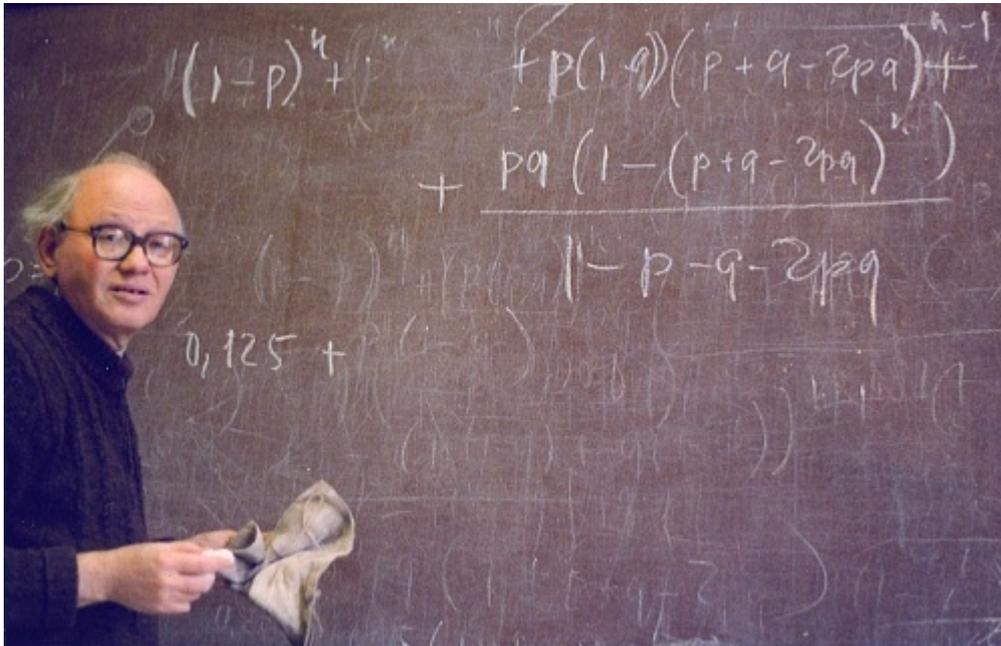
Complexités

	rechercher	ajouter	supprimer
Arbre binaire de recherche	$O(h)$	$O(h)$	$O(h)$

Souhait : On veut une hauteur h optimale

On veut un arbre équilibré

Arbres binaires de recherche de type AVL (Adelson-Velsky, Landis)

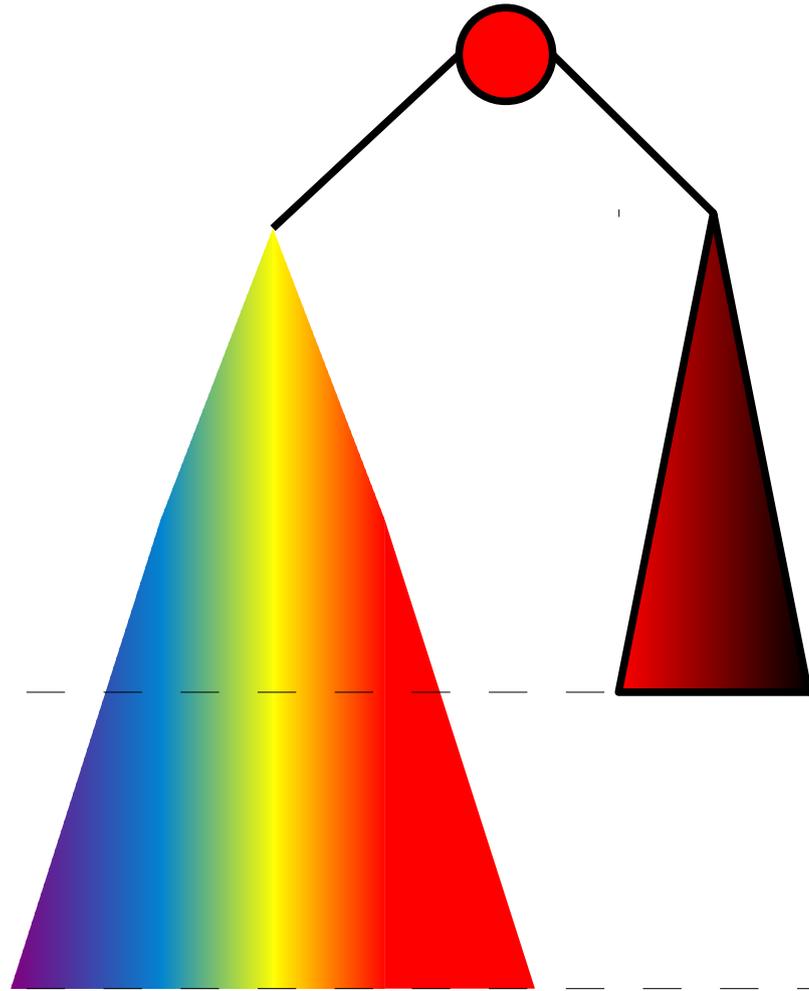


Адельсон-Вельский



Ландис

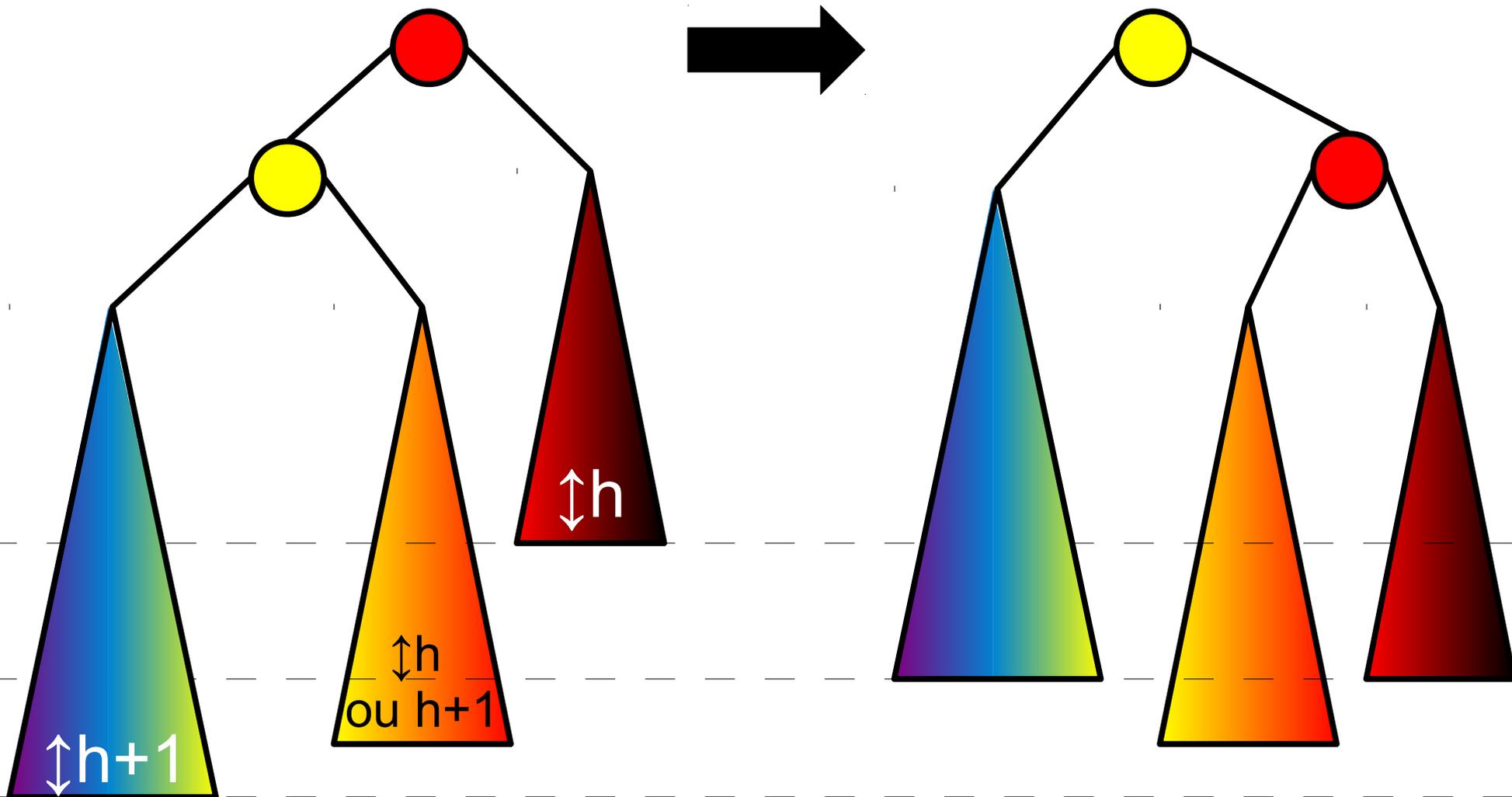
But : rééquilibrer



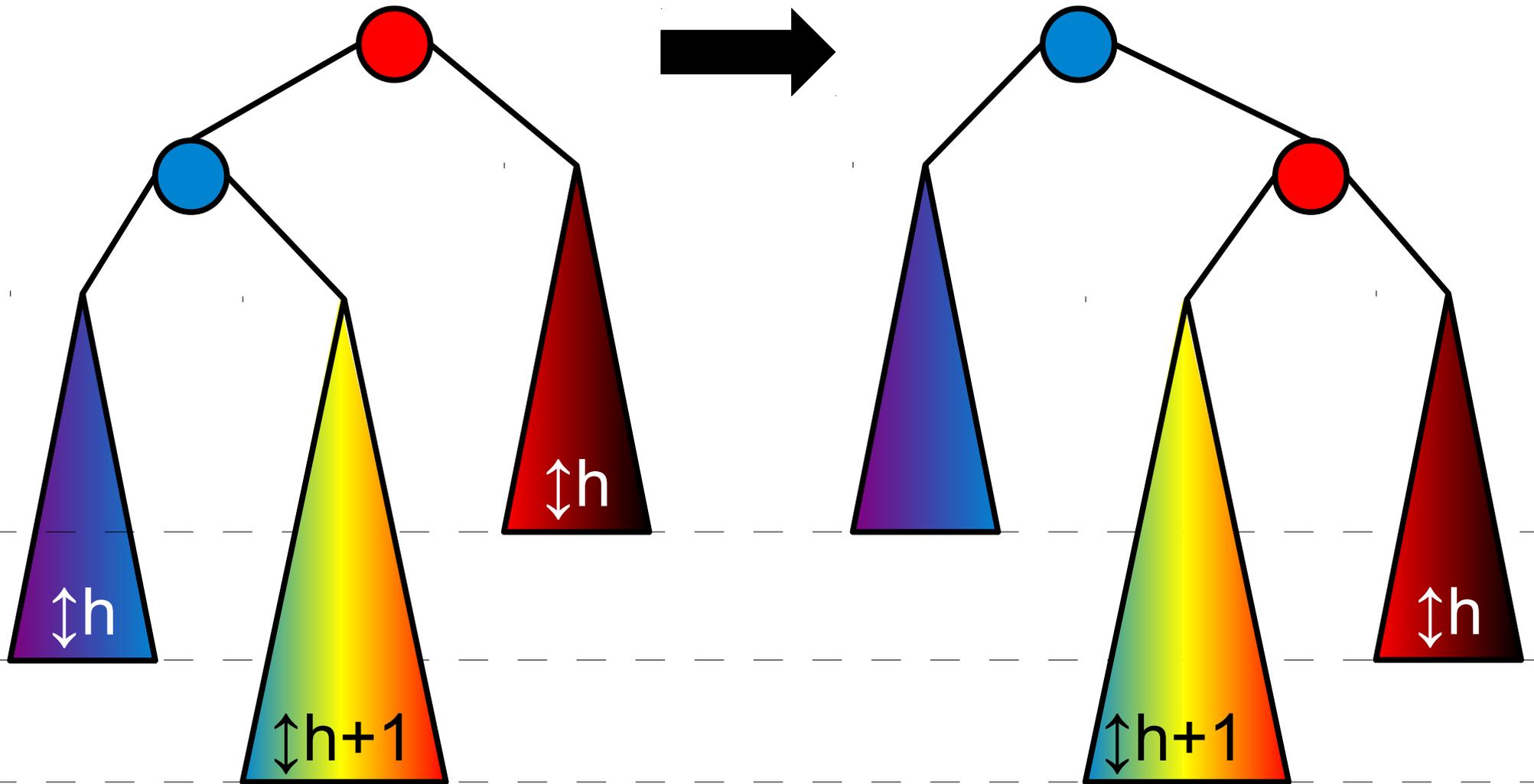
(ou son
symétrique)

2

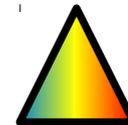
Rotation droite



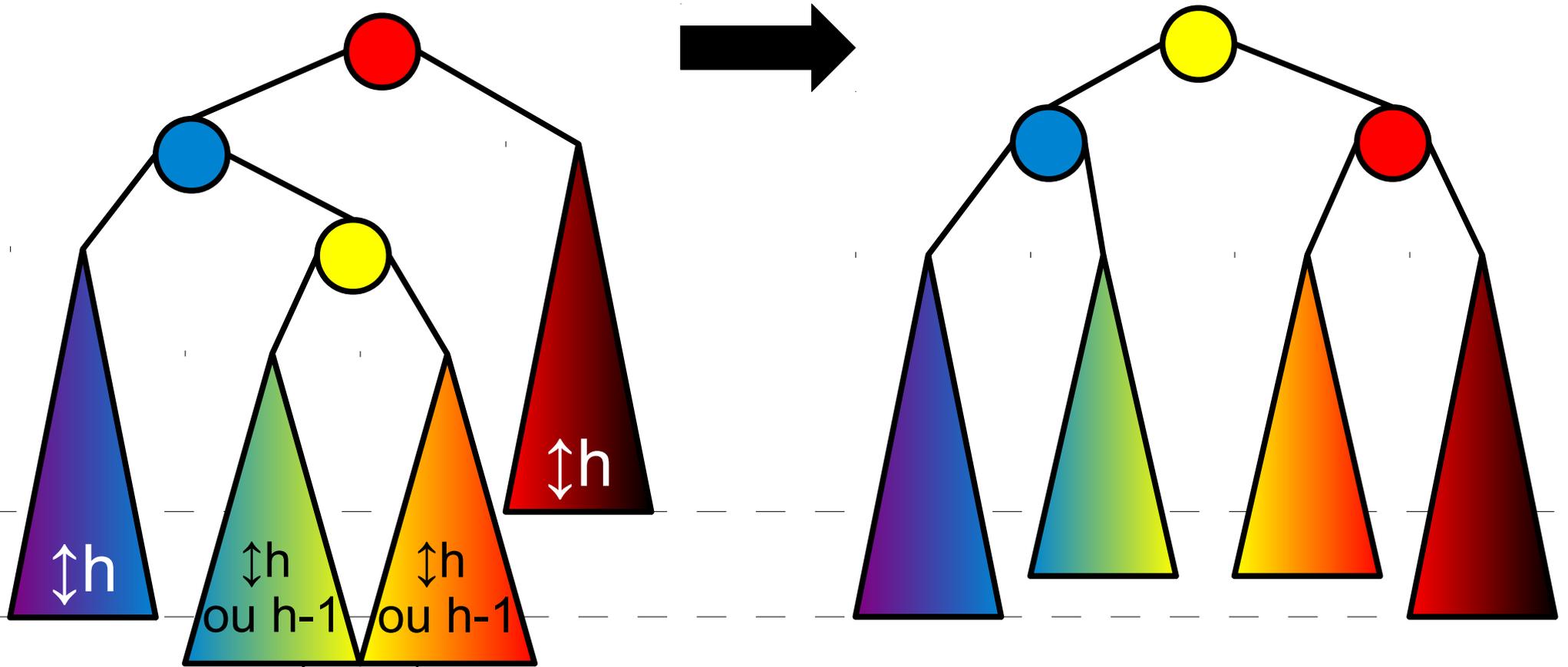
Essai d'une rotation droite !



Raté ! On va détailler

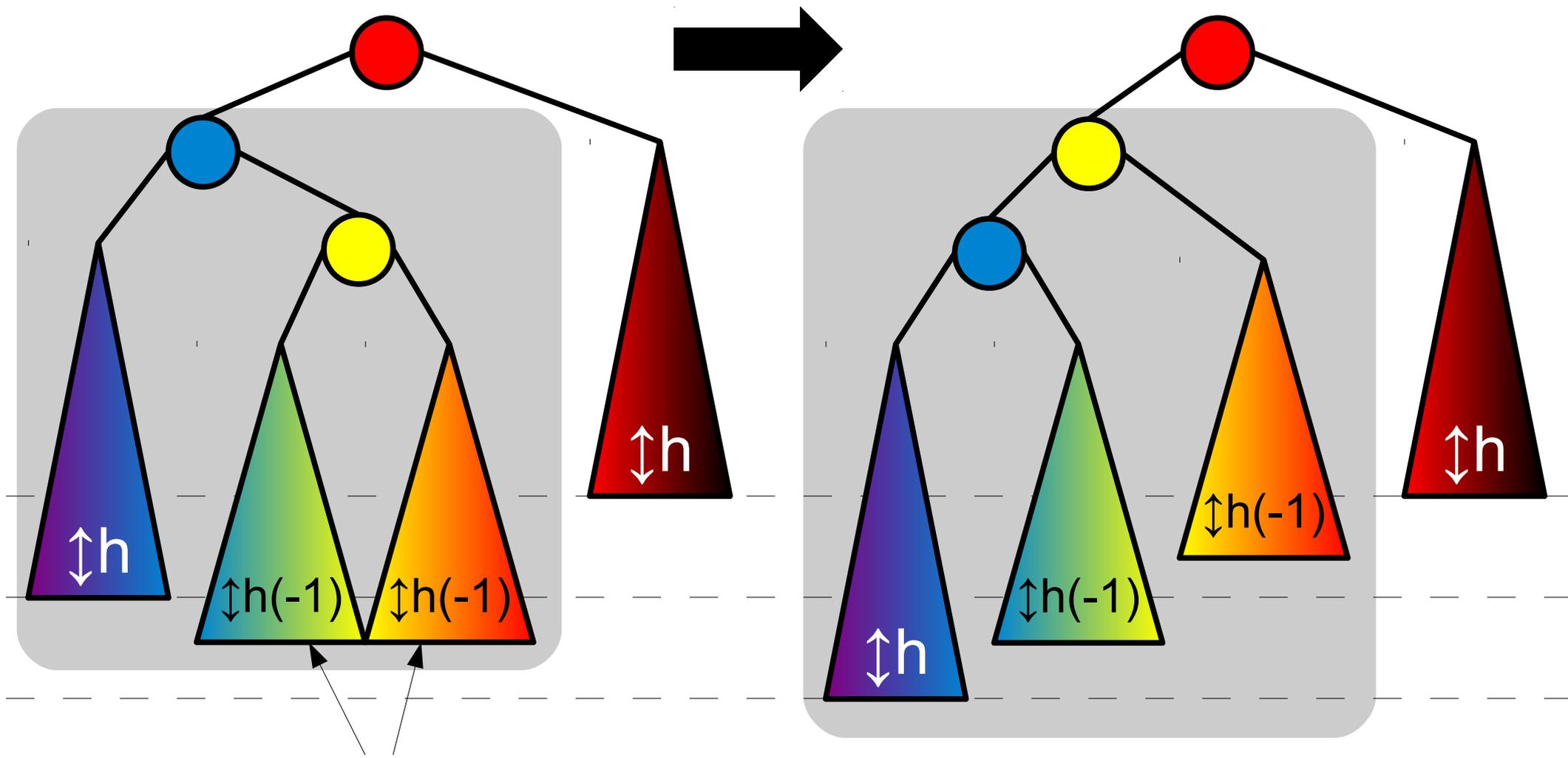


Rotation gauche-droite



L'un des deux de hauteurs h

D'abord une rotation gauche



L'un des deux de hauteurs h

Puis une rotation droite

