

Correction série algo 1a7

1. Donner trois exemples de problèmes pour lesquels on peut développer un algorithme.

- jouer au blackjack
- distributeur de boissons : gérer son fonctionnement :
 - gestion argent
 - distribution des produits
- rendu de monnaie

2. Pourquoi un algorithme de résolution d'un problème nous aide à écrire un programme pour résoudre ce problème à l'aide d'un ordinateur ?

Car quand on écrit un algorithme, on doit expliquer chaque étape de manière univoque et un ordinateur ne comprend que ce genre d'ordre (il ne sait pas interpréter une consigne ambiguë)

3. En 1990, le nombre de transistors sur un microprocesseur était 10^6 . D'après la loi de Moore, quel est ce nombre en : a. 1992 / b. 1994 / c. 2000 / d. 2021 ?

		année	nb de transistors
31 ans 15,5 x 2 ans	5 sauts de 2 ans 1 saut de 1 an	1990	10^6
		1992	$2 \cdot 10^6$
		1994	$4 \cdot 10^6$
		2000	$32 \cdot 10^6$
		2021	$4,63 \cdot 10^4 \cdot 10^6$ $= 4,63 \cdot 10^{10}$

$\times 2$
 $\times 2^5$
 $\times 2^{15,5} \left(\approx 46341 \right. \\ \left. \approx 4,63 \cdot 10^4 \right)$

4. Dites si chaque problème est *décisionnel* ou *de construction* (ou *d'évaluation*) :

a) Trouver le chemin le plus court entre deux points.

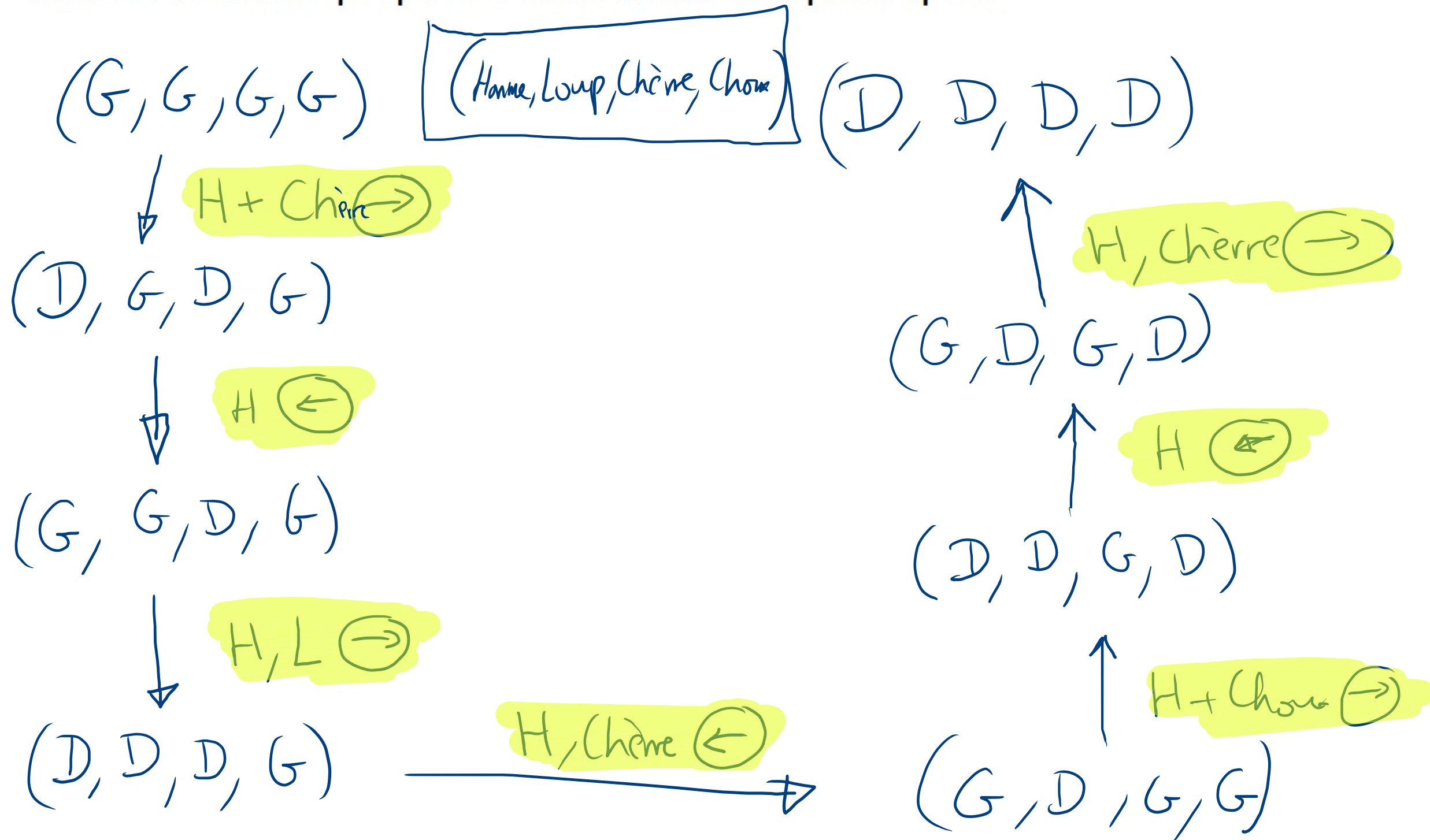
b) optimiser le rangement de cartons dans un dépôt,

c) trouver si un nombre donné est un nombre premier.

d) élaborer la grille horaire des trains dans une gare.

construction
construction
décisionnel
construction

5. Cherchez une solution au problème de la rivière.
Utilisez la force brute (envisager toutes les possibilités raisonnables)
Utilisez l'écriture proposée sous forme de quadruplet.



6. Expliquez à partir d'un exemple ce qu'est l'approche brutale (ou force brute) d'un problème algorithmique.

approche brutale: calculer toutes les possibilités envisageables
↓
choisir celle(s) qui sont optimales.

exemple: voyageur de commerce:
on calcule la distance totale pour tous les chemins (combinaison de villes) puis on cherche le résultat minimal → c'est le chemin optimal

7. Expliquez à partir d'un exemple en quoi le temps de calcul peut être une limitation importante dans la résolution d'un problème algorithmique.

Si on utilise un algorithme peu efficace (présumé)
et que la taille de son entrée est trop grande

⇒ temps gigantesque (par ex. plus que l'âge)
de l'univers



donc, on n'obtiendra pas la réponse!