Devoir Maison Algorithmique - Étienne Grandjean

Stéphane SECOUARD

24 octobre 2016

Table des matières

Exercice 1

Question 1 :

Une naissance correspond à *une épreuve de Bernoulli* en ce sens que deux résultat sont possibles :
— le succès : le nouveau né est une fille. Sa probabilité est $p = \frac{1}{2}$.
— l'échec : le nouveau né est un garçon. Sa probabilité est $q = \frac{1}{2}$.
On répète l'épreuve de Bernoulli de façon *indépendante*.

Soit X le nombre de *répétitions de l'épreuve de Bernoulli* qu'il faut faire pour rencontrer un succès : c'est la définition de la loi géométrique de paramètre $p = \frac{1}{2}$.

On peut alors calculer l'epérance de X qui donne le nombre d'enfants à avoir en moyenne pour avoir une fille : $\underline{E(X) = \frac{1}{p}} = \dfrac{1}{\frac{1}{2}} = \underline{2}$.

Cependant, nous cherchons à établir le nombre moyen de garçons obtenu avant d'avoir une fille.

Comme l'espérance de X donne le nombre moyen d'enfants (donc en comptant aussi la fille) on peut donc en déduire que **le nombre moyen de garçons obtenu avant d'avoir une fille est de 1.**

Question 2 :

Cette situation ressemble à la précédente si ce n'est qu'on s'arrête quoi qu'il arrive au bout de k étapes.

Ainsi les probabilités associées sont les mêmes jusqu'à l'étape $k-1$. Plus précisemment, si X désigne le nombre d'étapes à réaliser pour stopper le processus (ne plus faire d'enfant), on a :

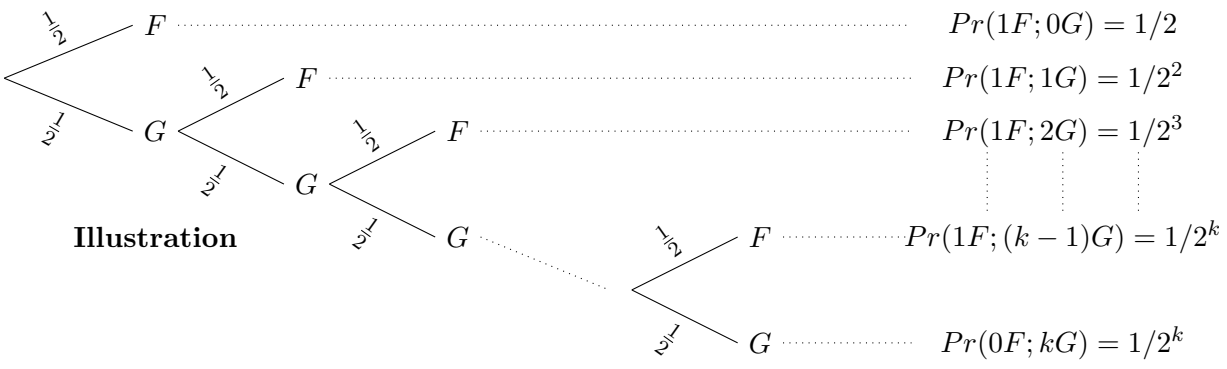
— $P(X = l) = pq^{l-1}$ pour $1 \leqslant l \leqslant k-1$ donc comme $p = q = \frac{1}{2}$, $P(X = l) = \dfrac{1}{2^l}$

— $P(X = k) = pq^{k-1} + q^k$ qui donne pour $p = q = \frac{1}{2}$, $P(X = k) = 2p^k = \dfrac{1}{2^{k-1}}$.

Mais ce qui nous intéresse pour cette question est plutôt de savoir combien de fille et de garçons nous avons dans chaque cas.

Or si $P(X = l)$ avec $1 \leqslant l \leqslant k-1$ correspond bien à la probabilité d'avoir 1 fille et $l-1$ garçons, $P(X = k)$ ne donne que la probabilité d'avoir k enfants (1fille et k-1 garçons ou k garçons). C'est pourquoi, comme ces deux issues sont équiprobables, on obtient les probabilités de chacune de ces deux issues en divisant par 2 (on obtient alors $\frac{1}{2^k}$).

Ainsi :
— la probabilité d'avoir une fille et 0 garçon est 1/2
— la probabilité d'avoir une fille et 1 garçon est $1/2^2$
— la probabilité d'avoir une fille et 2 garçons est $1/2^3$
…
— la probabilité d'avoir une fille et k-2 garçons est $1/2^{k-1}$
— la probabilité d'avoir une fille et k-1 garçons est $1/2^k$
la dernière ligne ne suit pas la même régularité :
— la probabilité d'avoir k garçons (sans fille) est $1/2^k$



Illustration

Calcul des moyennes :

pour les filles : Si F désigne la variable aléatoire donnant le nombre de filles, comme il n'y a que 1 fille ou pas de fille, il est simple de voir que l'on a :

$$\begin{aligned} E(F) &= P(F=0)\times 0 + P(F=1)\times 1 \\ &= P(F=1) \\ &= 1 - P(F=0) \\ &= 1 - \frac{1}{2^k} \end{aligned}$$

pour les garçons : Si G désigne la variable aléatoire donnant le nombre de garçons, compte-tenu des probabilités données précédemment, on a :

$$\begin{aligned} E(G) &= P(G=0)\times 0 + P(G=1)\times 1 + \ldots + P(G=k-1)\times(k-1) + P(G=k)\times k \\ &= \sum_{i=0}^{k-1} \frac{i}{2^{i+1}} + \frac{k}{2^k} \\ &= \sum_{i=0}^{k-1} \frac{i}{2^{i+1}} + \frac{k}{2^{k+1}} + \frac{k}{2^{k+1}} \\ &= \sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k}{2^{k+1}} \end{aligned}$$

Après quelques essais numériques j'ai constaté que cette valeur était la même que celle obtenue pour l'espérance du nombre de filles.

Je vais donc montrer par récurrence que mes résultats numériques se généralisent (je n'ai pas trouvé plus élégant) :

Posons $P(k)$: « $\sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k}{2^{k+1}} = 1 - \frac{1}{2^k}$ »

Initialisation : D'une part $\sum_{i=0}^{1} \frac{i}{2^{i+1}} + \frac{1}{2^{1+1}} = \frac{1}{2^2} + \frac{1}{2^2} = \frac{1}{2}$

D'autre par $1 - \frac{1}{2^1} = \frac{1}{2}$ donc $P(1)$ est vraie.

Hérédité : Soit $k \geqslant 1$.

Supposons que $P(k)$ est vraie.

$$\begin{aligned} \sum_{i=0}^{k+1} \frac{i}{2^{i+1}} + \frac{k+1}{2^{k+1+1}} &= \sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k+1}{2^{k+2}} + \frac{k+1}{2^{k+2}} \\ &= \sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k+1}{2^{k+1}} \\ &= \sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k}{2^{k+1}} + \frac{1}{2^{k+1}} \\ &= 1 - \frac{1}{2^k} + \frac{1}{2^{k+1}} \text{ C'est l'hypothèse de récurrence} \\ &= 1 - \frac{2}{2^{k+1}} + \frac{1}{2^{k+1}} \\ &= 1 - \frac{1}{2^{k+1}} \end{aligned}$$

$P(k+1)$ est vraie.

Conclusion : $P(1)$ est vraie et, pour tout $k \geqslant 1$, si $P(k)$ est vraie alors $P(k+1)$ aussi.

On a donc prouvé par récurrence que $P(k)$ est vraie pour tout $k \geqslant 1$:

$$\boxed{\forall k \geqslant 1 \ , \ \sum_{i=0}^{k} \frac{i}{2^{i+1}} + \frac{k}{2^{k+1}} = 1 - \frac{1}{2^k}}$$

On a donc prouvé que dans ce cas :

le nombre moyen de filles obtenu est le même que celui de garçons : dans chaque cas $1 - \frac{1}{2^{k+1}}$

Que le nombre d'enfants soit en moyenne moins élevé qu'au cas précédent semble logique puisqu'on s'autorise moins d'enfants. Par contre que les deux moyennes soient les mêmes pour les filles et les garçons est assez contre intuitif car on pourrait penser que le nombre de filles "non faites" ne compense pas le nombre de garçons mais ce n'est pas le cas.

Exercice 3 :

Question 1 :

Soient trois entiers a, p et $n \geqslant 1$ tels que $a < n$. Voici un algorithme qui calcule l'entier a^p mod n.

```
1 | resultat=1 , puissance=a, w=p
2 | tant que w>0 :
3 |     si w=1 [mod 2]
4 |     alors resultat=puissance × resultat [mod n]
5 |     puissance=puissance × puissance [mod n]
6 |     w=w//2 (division entière)
7 | renvoyer resultat
```

Question 2 :

- p s'écrit en binaire en utilisant la décomposition en base 2 de p, plus précisemment, si $2^{l-1} \leqslant p < 2^l$ alors on a $p = \sum_{k=0}^{l-1} p_k \times 2^k$.
  Ainsi la taille de l'écriture en binaire sera l.
  En appliquant le log_2 à l'inégalité précédente on a : $l - 1 \leqslant log_2(p) < l$.
  On peut donc en déduire que $l = \lfloor log_2(p) \rfloor + 1$.
- Dans l'algorithme chaque tour de boucle correspond à l'exploration d'un bit de p(puisqu'on divise par deux à chaque étape, ce qui revient à l'exploration des bits de p de droite à gauche).
  Ainsi il y a exactement l tours de boucle (puisque c'est la taille du développement binaire de p).
  Dans la boucle, il y a deux multiplications modulaires : celle de la ligne 5 qui s'effectue systématiquement et celle de la ligne 4 qui s'effectue si et seulement si le bit de p étudié est 1.
  Le nombre de multiplications modulaires est de l + nombre de 1 dans l'écriture en binaire de p.
- Ainsi, si p ne contient que des 1, alors il y a $2l$ multiplications et si p est une puissance de 2 il y a exactement $l + 1$ multiplications. C'est l'intervalle du nombre de multiplications pour p de longueur l.
- Le nombre exacte de multiplications est exactement $l + u$ où l est la longueur de p et u le nombre de bits 1 de p.

Question 3 :

- les lignes 4 et 7 sont négligeables en complexité par rapport aux multiplications car effectuer le modulo 2 revient à lire le bit de poids faible tandis que la division entière par 2 consiste à enlever le bit de poids faible.
- On peut donc dire que la complexité de l'algorithme correspond aux calculs des multiplications (qui sont entre l et $2l$ donc en $O(l)$).
- Comme chaque multiplication s'effectue en $O(l \log l)$, on en déduit que la complexité de cette algorithme est en $O(l^2 \log l)$.
- Pour $l \approx 10\,000$, on obtient une complexité en temps de l'ordre du milliard.
- en multipliant la taille l par 2, la complexité en temps augmente d'une valeur de l'ordre de $4 \log 2 \approx 3$.

Question 4 :

Voici l'algorithme réalisé sous python qui implémente la fonction expo(a,p,n) qui retourne $a^n[mod\, p]$ et le nombre de multiplications effectuées :

```
def expo(a,p,n) :
    resultat,puissance,w,mult=1,a,p,0
    while w>0 :
        if w%2==1 :
            resultat=puissance*resultat%n
            mult+=1
        puissance=puissance*puissance%n
        mult+=1
        w=w//2
    return (resultat,mult)
```

Question 5 :

J'ai ajouté à la suite du code donné à la question 4 ceci :

```
def genere(borneinf,bornesup) :
    l=[ ]
    for k in range(1000) :
        l.append(randint(borneinf,bornesup))
    return l
if ___name___=='___main___' :
    a,n=123,987
    l=genere(2**19,2**20-1)
    compteur=0
    for k in l :
        compteur+=expo(a,k,n)[1]
    moyenne=compteur/len(l)
    print(moyenne)
```

On obtient comme moyenne 30 (je n'ai pas eu d'autres valeurs dans mes essais).

C'est logique car on a vu que le nombre de multiplications est donné par $l + u$. Or ici $l = 20$ car les valeurs de p sont prises entre 2^{19} et $2^{20} - 1$. De plus, comme ces valeurs sont des générations aléatoires de 20 bits, en moyenne, il y a 10 fois le bit 1 et 10 fois le bit 0 : cela donne en moyenne une valeur pour u de 10.

On retrouve bien en moyenne 20+10=30.

Exercice 4

Question 1 :

Rappel du théorème :

Théorème (Schwartz et Zippel)**.**

- *Soit K un corps quelconque et un ensemble fini $S \subseteq K$ quelconque.*
- *Soit un polynôme $P(x_1, \ldots, x_n) \in K[x_1, \ldots, x_n]$ non nul de degré total d.*
- ***L'équation** $P(x_1, \ldots, x_n) = 0$ **a au plus** $d.|S|^{n-1}$ **solutions dans** S^n.*

cas d'une seule variable : le théorème adapté à une seule variable donne que « l'équation $P(x) = 0$ a au plus $d.|S|^0 = d$ solutions dans S.

- Or un polynôme de degré d a au plus d racines dans K (la preuve repose sur une récurrence sur le degré du polynôme et sur la division euclidienne qui est possible dans le cas d'une seule variable car $K[x]$ est euclidien puisque K est un corps).
- Donc P a au plus d racines dans K donc a au plus d racines dans $S \subseteq K$. C'est bien ce qu'il fallait démontrer.

cas d'un degré $d = 1$: Il s'agit de prouver que $P(x_1, \ldots, x_n) = 0$ a au plus $|S|^{n-1}$ solutions dans S^n (puisque ici $d = 1$).

— Or si $d = 1$ c'est que P est une somme de monômes de degré 1. Plus précisemment, P peut s'écrire $P(x_1, \ldots, x_n) = a_1x_1 + a_2x_2 + \ldots + a_nx_n$ avec $a_1, \ldots, a_n \in K$.

— Mais on peut prouver que pour tout (n-1) uplets $(x_2, \ldots, x_n) \in S^{n-1}$, il existe un et un seul $x_1 \in K$ tel que $P(x_1, x_2, \ldots, x_n) = 0$.
En effet,

$$\begin{aligned} P(x_1, x_2, \ldots, x_n) = 0 &\iff a_1x_1 + a_2x_2 + \ldots + a_nx_n = 0 \\ &\iff a_1x_1 = -a_2x_2 - a_3x_3 - \ldots - a_nx_n \\ &\iff x_1 = a_1^{-1}(-a_2x_2 - a_3x_3 - \ldots - a_nx_n) \end{aligned}$$

— Autrement dit, pour tout (n-1) uplets de $(x_2, \ldots, x_n) \in S^{n-1}$, il existe **au plus** un $x_1 \in \mathbf{S} \subseteq K$ tel que $P(x_1, x_2, \ldots, x_n) = 0$.

— Comme le nombre de (n-1) uplets de S^{n-1} est $|S|^{n-1}$, il existe au plus $|S|^{n-1}$ n-uplets de S^n solutions de $P(x_1, \ldots, x_n) = 0$.

Question 2 :

Test (Test de Schwartz et Zippel :)**.** *Soit P de degré d. On fixe $S \subseteq K$.*

Répéter k fois
  Choisir au hasard $(s_1, \ldots, s_m) \in S^m$
  Calculer $P(s_1, \ldots, s_m)$
  Si $P(s_1, \ldots, s_m) \neq 0$
  Alors retourner NON (Il est certain que P n'est pas nul)
retourner OUI (il est probable que P soit nul)

Je n'ai pas précisé les précautions du cours car la question invite à une discussion sur ce sujet :

— si le test dit que P est non nul il est absolument certain que c'est le cas !

— si le test dit que P est nul ce n'est pas absolument certain, en effet :

- Si P n'est pas nul alors la probabilité que $P(s_1, \ldots, s_m) = 0$ pour un m-uplet pris au hasard dans S est majorée par $\frac{\frac{d}{|S|^{n-1}}}{|S|^n} = \frac{d}{|S|}$.
  Autrement dit la probabilité d'affirmer que P est nul car un m-uplet de S pris au hasard l'annule et de se tromper est majorée par $\frac{d}{|S|}$.
- En répétant k fois cette expérience (ce que fait le test), on peut dire que la probabilité d'affirmer que P est nul car les k m-uplets l'annule et de se tromper est majorée par $\left(\frac{d}{|S|}\right)^k$.
- L'erreur est donc majorée par cette quantité.

— Pour que ce test fonctionne il faut
  — prendre un ensemble S assez grand et surtout de cardinal strictement plus grand que d ! Dans le cours il a été pris 2d pour assurer une division par 2 de l'erreur à chaque tour.
  — réaliser un assez grand nombre de répétitions k pour assurer un phénomène d'amplification au test.

— Pour réaliser une erreur inférieur à 10^{-15}, on peut par exemple prendre un ensemble S de cardinal au moins égal à $10d$ et réaliser au moins $k = 15$ répétitions.
  L'erreur serait alors inférieur à $\left(\frac{d}{10d}\right)^{15} = 10^{-15}$.

Question 3 :

« $P_1 = P_2 \iff P_1 - P_2$ nul »donc il suffit de tester si le polynôme $P_1 - P_2$ est nul :

— si le test dit que le $P_1 - P_2$ n'est pas nul on peut affirmer avec certitude que $P_1 \neq P_2$.

— si le test dit que $P_1 - P_2$ on peut dire qu'il est probable (avec les précautions dont nous avons discutées précédemment) que $P_1 = P_2$.

Question 4 :

A a trois fils : $r(f)$, $r(r(f))$ et $r(f, r(f, f, f))$. Nous allons d'abord donner les polynômes de chacun d'entre eux :

- f a pour polynôme x_0 donc $r(f)$ a pour polynôme $x_1 - x_0$.
- $r(f)$ a pour polynôme $x_1 - x_0$ donc $r(r(f))$ a pour polynôme $x_2 - (x_1 - x_0)$.
- $r(f, f, f)$ a pour polynôme $(x_1 - x_0)(x_1 - x_0)(x_1 - x_0)$ donc le polynôme de $r(f, r(f, f, f))$ est $(x_2 - x_0)(x_2 - (x_1 - x_0)(x_1 - x_0)(x_1 - x_0))$.

On peut avec ces polynômes générer le polynôme de A :

$$p_A(x_0, x_1, x_2, x_3) = (x_3 - (x_1 - x_0))(x_3 - (x_2 - (x_1 - x_0)))(x_3 - (x_2 - x_0)(x_2 - (x_1 - x_0)(x_1 - x_0)(x_1 - x_0)))$$

Question 5 :

Faisons une induction sur la hauteur des sous-arbres d'un arbre A donné :

- Au niveau des feuilles (hauteur 0), le polynôme est x_0 qui est de degré 1 et 1 est évidemment le nombre de feuille d'une feuille !
- Supposons qu'à une certaine hauteur $h-1$ le degré des polynômes associés aux sous arbres soit égale à celui du nombre de feuilles.
  Ainsi, nous allons prouver le résultat attendu sur $r(A_1, \ldots, A_k)$ de hauteur h en supposant qu'il est vérifié sur les A_i pour $i \in [1; k]$ (qui sont donc au plus de hauteur $h-1$ car fils d'un arbre de hauteur h) :
  - les polynômes p_{A_i} sont au moins de degré 1 (puisqu'ils contiennent au moins une feuille) et ne contiennent pas la variable x_h (car sont de hauteur au plus $h-1$).
    On peut donc en déduire qu'*un polynôme $x_h - p_{A_i}$ est du même degré que p_{A_i}* car il n'y a pas de monômes en x_h dans p_{A_i} (ce qui évite de faire disparaître un monôme de p_{A_i} ce qui aurait pu faire baisser le degré) et que le degré de p_{A_i} est au moins égal à celui de x_h (car au moins de degré 1).
  - Comme $p_{r(A_1,\ldots,A_k)}$ est le produit des polynômes de la forme $x_h - p_{A_i}$, son degré est la somme des degrés de ces polynômes, c'est à dire en utilisant l'hypothèse de l'induction,
    $\sum_{i=1}^{k} degr(A_i) = \sum_{i=1}^{k}$(nombre de feuilles de A_i). Or la somme des nombres de feuilles des A_i est exactement le nombre de feuille de $p_{r(A_1,\ldots,A_k)}$. C'est ce qu'on cherchait à établir.

Pour tout arbre A, on a bien prouvé par induction que le degré de p_A est bien égal à son nombre de feuilles.

Question 6 :

Faisons une induction sur la hauteur des sous-arbres d'un arbre A donné :

- Au niveau des feuilles (taille 0 par définition), on a bien : taille(une feuille)+nombreFeuilles(une feuille)=0+1=1=taille(x_0).
  Et comme x_0 est le polynôme associé à une feuille, on a bien le résultat attendu.
- Supposons qu'à une certaine hauteur $h-1$ la taille des polynômes associés aux sous arbres soit égale à la somme de leur taille et de leur nombre de feuilles.
  Ainsi, nous allons prouver le résultat attendu sur $r(A_1, \ldots, A_k)$ de hauteur h en supposant qu'il est vérifié sur les A_i pour $i \in [1; k]$ (qui sont donc au plus de hauteur $h-1$ car fils d'un arbre de

hauteur h) :

$$\begin{aligned}
taille(p_{r(A_1,\ldots,A_k)}) &= taille(\prod_{i=1}^{k}(x_h - p_{A_i}) \\
&= \sum_{i=1}^{k}(taille(x_h - p_{A_i})) \\
&= \sum_{i=1}^{k}(1 + taille(p_{A_i})) \\
&= \sum_{i=1}^{k} 1 + \sum_{i=1}^{k}(taille(A_i) + nombreFeuilles(A_i)) \\
&= k + \sum_{i=1}^{k} taille(A_i) + \sum_{i=1}^{k} nombreFeuilles(A_i) \\
&= taille(r(A_1,\ldots,A_k)) + nombreFeuilles(r(A_1,\ldots,A_k))
\end{aligned}$$

On a bien prouvé que la taille du polynôme associé à $r(A_1,\ldots,A_k)$ est égale à la somme de la taille de $r(A_1,\ldots,A_k)$ et de son nombre de feuilles.

Pour tout arbre A, on a bien prouvé par induction que la taille de p_A est bien égal à taille(A)+nombreFeuilles(A).

La taille d'un arbre est exactement le nombre de liens parent-enfant car à chaque fois qu'on remonte dans la hiérarchie on rajoute à la taille le nombre de liens du parent introduit avec chacun de ses enfants.

Donc, si on prend le vocabulaire des graphes, la taille d'un arbre est donc le nombre de ses arêtes. Or le nombre des arêtes d'un graphe est exactement égal à son nombre de noeuds moins 1 (j'admets ce résultat mais cela peut se montrer par induction).

De plus, le nombre de feuilles est au plus égale au nombre de noeuds (puisqu'une feuille est un noeud).

Ainsi, pour un arbre A donné, la taille de p_A qui est égale à taille(A)+nombreFeuilles(A) est inférieur à 2 fois le nombre de noeuds (puisque chaque élément de la somme est inférieur au nombre de noeuds).

Cela montre que $\boxed{taille(p_A) = O(nombreNoeuds(A))}$.

(On peut même dire que $taille(p_A) = \Theta(nombreNoeuds(A))$ car la taille de l'arbre vaut le nombre de noeuds de A).

Question 7 :

— Deux arbres de hauteur 0 sont toujours isomorphes par définition.
Ils ont aussi la même hauteur (0), la même taille (0), le même nombre de feuilles (1) et des polynômes identiques (x_0). Donc pour deux arbres de hauteur 0, l'équivalence annoncée est bien vérifiée.

— Soient deux arbres, de hauteurs non nulles, $A = r(A_1,\ldots,A_k)$ et $B = r(B_1,\ldots,B'_k)$

$$\begin{aligned}
A \simeq B \iff & \quad k = k' \text{ et } \exists \sigma \in S_k \text{ tel que } \forall i \in [1;k] A_i \simeq B_{\sigma(i)}. \\
\iff & \quad k = k' \text{ et } \exists \sigma \in S_k \text{ tel que } \forall i \in [1;k] \begin{cases} hauteur(A_i) = hauteur(B_{\sigma(i)}) \\ taille(A_i) = taille(B_{\sigma(i)}) \\ nombreFeuilles(A_i) = nombreFeuilles(B_{\sigma(i)}) \\ p_{A_i} = p_{B_{\sigma(i)}} \end{cases}
\end{aligned}$$

$\Rightarrow$ **:** Supposons que $A \simeq B$ (on a alors les conditions équivalentes citées ci-dessus).
On a alors (la justification essentielle est que σ est une bijection de $[1;k]$ vers $[1;k]$ et donc il revient au même de parcourir tous les i ou tous les $\sigma(i)$) :

$$\begin{aligned} hauteur(A) &= 1 + \underset{i\in[1;k]}{Max}(hauteur(A_i)) \\ &= 1 + \underset{i\in[1;k]}{Max}(hauteur(B_{\sigma(i)})) \\ &= 1 + \underset{i\in[1;k]}{Max}(hauteur(B_i)) \\ &= hauteur(B) \end{aligned}$$

$$\begin{aligned} taille(A) &= k + \sum_{i=1}^{k} taille(A_i) \\ &= k + \sum_{i=1}^{k} taille(B_{\sigma(i)}) \\ &= k + \sum_{i=1}^{k} taille(B_i) \\ &= taille(B) \end{aligned}$$

$$\begin{aligned} nbFeuilles(A) &= \sum_{i=1}^{k} nbFeuilles(A_i) \\ &= \sum_{i=1}^{k} nbFeuilles(B_{\sigma(i)}) \\ &= \sum_{i=1}^{k} nbFeuilles(B_i) \\ &= nbFeuilles(B) \end{aligned}$$

$$\begin{aligned} p_A &= \prod_{i=1}^{k}(x_h - p_{A_i}) \\ &= \prod_{i=1}^{k}(x_h - p_{B_{\sigma(i)}}) \\ &= \prod_{i=1}^{k}(x_h - p_{B_i}) \\ &= p_B \end{aligned}$$

A et B vérifient bien toutes les conditions données par l'énoncé. On a donc prouvé l'implication en partant de $A \simeq B$.

$\Leftarrow$ **:** Supposons que $p_A = \prod_{i=1}^{k}(x_h - p_{A_i}) = p_B = \prod_{i=1}^{k'}(x_h - p_{B_i})$.

Comme la factorisation d'un polynôme est unique à l'ordre des facteurs près, on a forcément $k = k'$ et les k facteurs de p_A sont identiques aux k facteurs de p_B à permutations près.

Autrement dit, il existe une permutation π de l'ensemble des indices $\{1, \ldots, k\}$ telle que $p_{A_i} = p_{B_{\pi(i)}}$ pour tout indice $i \in \{1, \ldots, k\}$. Mais par hypothèse d'induction, dire que $p_{A_i} = p_{B_{\pi(i)}}$ revient à dire que $A_i \simeq B_{\pi(i)}$.

On a bien prouvé que $k = k'$ et qu'il existe une permutation π de l'ensemble des indices $\{1, \ldots, k\}$ telle que $A_i \simeq B_{\pi(i)}$ pour tout indice $i \in \{1, \ldots, k\}$: ainsi $A \simeq B$.

Les deux implications permettent d'affirmer que $A \simeq B$ si et seulement si les conditions suivants sont toutes deux réalisées :

1. A et B sont de même hauteur h et de même taille et ont le même nombre de feuilles d ;
2. les polynômes associés p_A et p_B sont identiques.

(Ça ne bloque pas le raisonnement mais j'ai tout de même l'impression que la condition 2 implique la condition 1, ce qui rend cette dernière inutile dans ma démonstration. Ou bien quelque chose m'a échappé)

Question 8 :

Soient deux arbres A et B.

Je suppose qu'il existe une fonction hauteur, taille, nbFeuilles et poly qui donnent respectivement la hauteur, la taille, le nombre de feuilles et le polynôme associé à un arbre. Je suppose également que nous disposont d'un corps K et d'une fonction ensAkElements(n) capable de générer un ensemble de n éléments de K.

```
1  Si | hauteur(A) ≠ hauteur(B) ou si nbFeuilles(A) ≠ nbFeuilles(B)
2     | ou si taille(A) ≠ taille(B)                                    (1)
3  Alors retourner A et B ne sont pas isomorphes (résultat certain)
4  d=nbFeuilles(A)
5  S=ensAkElement(2d)                                                 (2)
6  Répéter k fois                                                     (3)
7     | Choisir au hasard (s_1, ..., s_d) ∈ S^d
8     | Calculer P_A(s_1, ..., s_d) − P_B(s_1, ..., s_d)
9     | Si P_A(s_1, ..., s_d) − P_B(s_1, ..., s_d) ≠ 0
10    | Alors retourner A et B ne sont pas isomorphes (c'est certain)
11 retourner A et B sont isomorphes (c'est probable)
```

(1) Cette étape permet de faire un test élémentaire de la situation et me permet de ne manipuler qu'une taille, qu'une hauteur et qu'un nombre de feuilles ici et dans la question d'après.

(2) on a prouvé à la question 5 que le degré de p_A est égale à son nombre de feuilles et on a vu qu'en fixant un ensemble deux fois plus grand que le degré du polynôme à tester on divisait (au moins) par 2 la probabilité d'erreurs à chaque tour de boucle.

(3) k est à choisir en fonction de la marge d'erreur qu'on veut se donner : l'erreur possible est ici que l'algorithme prétende que A et B sont isomorphes alors qu'ils ne le sont pas. Cette erreur a une probabilité inférieure à $\frac{1}{2^k}$ de se produire (c'est l'analyse faite à la question 2).

Question 9 :

je suppose donné la hauteur des arbres et leur nombre de feuilles (car j'ignore comment ceci est calculé). On se place dans le cas où la complexité sera la plus grande (l'algorithme prétendant que A et B sont isomorphes)

l1→ l4	les 3 premières lignes de l'algorithme sont négligeables par rapport au reste	
l5	Générer S se fait par génération de 2d nombres aléatoire :	**2d**
l7	Choisir au hasard $(s_1, \ldots, s_d)$ se fait par génération de d nombres aléatoires et cette étape est faite k fois :	**+k.d**
l8	Calculer $P_A(s_1, \ldots, x_h)$ utilise un nombre d'opérations égales à la taille du polynôme moins 1 car il y a exactement une opération à effectuer entre chacune des variables. Comme nous devons aussi faire ce calcul pour P_B et ensuite soustraire les résultats, on doit faire $taille(p_A)-1+taille(p_B)-1+1$. De plus, comme la taille du polynôme est donné par la somme de la taille de l'arbre et de sa hauteur, comme les tailles et hauteurs des deux arbres sont les mêmes (la première ligne de l'algorithme écarte les autres cas), on peut dire que le nombre d'opérations est : $taille(A)+hauteur(1)-1+taille(A)+hauteur(1)-1+1 = 2taille(1)+2hauteur(A)-1$:	**+2k.taille(A) +2k.hauteur(A) -k**
l9	une soustraction (répétée k fois) :	**+k**
	TOTAL : 2d+k.d+2k.taille(A)+2k.hauteur(A)+k	

Cette complexité est peu parlante mais :

— k est une valeur fixée (par exemple avec k=40 on aura une probabilité d'erreur inférieur à $\frac{1}{2^{40}}$).

— la hauteur est inférieur à la taille de l'arbre.

— le nombre de feuilles d est comparable ou inférieur à la taille.

Donc dans la complexité est majorable par une constante multipliée par la taille de l'arbre : l'algorithme a une complexité en $O(taille\,de\,l'arbre)$.