TD - RECURSIVITE
Exponentiation rapide

Corrigé

Introduction et objectifs

Dans ce TD consacré à la récursivité, on étudie un algorithme permettant le calcul de a^n où a est un réel non nul et n un entier naturel. L'objectif principal du TD est une étude, à la fois expérimentale et théorique, de la complexité de l'algorithme proposée.

Code Python

```
def fexpo(a,n):
     if n == 0:
          return 1
     else:
          q,r = divmod(n,2)
          prod = fexpo(a,q)
          prod = prod * prod
          if r == 1:
               prod = prod * a
          return prod
```

Remarque : la fonction « `divmod` » reçoit comme arguments deux entiers, α et β, et renvoie deux autres entiers, q et r, correspondant respectivement au quotient et au reste de la division euclidienne de α par β.

1. Expliquez le principe de l'algorithme ayant conduit à cette fonction Python.

Lorsque l'entier n est non nul, on effectue sa division euclidienne par 2 (instruction `q,r = divmod(n,2)`) puis on effectue un appel récursif à la fonction `fexpo` avec les arguments `a` et `q`. Le résultat obtenu (`prod`) est alors multiplié par lui-même et, lorsque `r` est égal à 1, à nouveau par `a`.
L'algorithme implémente ainsi le calcul : $a^n = a^{2q+r} = a^q \times a^q \times a^r$.
C'est bien le calcul que l'on cherche à effectuer.

2. Quelles sont les principales étapes du calcul de 2^{53} ? Combien de multiplications sont requises pour ce calcul ? Combien de multiplications sont requises dans le calcul « naïf » ?

On a :

$$2^{53} = 2^{26} \times 2^{26} \times 2$$
$$2^{26} = 2^{13} \times 2^{13}$$
$$2^{13} = 2^{6} \times 2^{6} \times 2$$
$$2^{6} = 2^{3} \times 2^{3}$$
$$2^{3} = 2^{1} \times 2^{1} \times 2$$
$$2^{1} = 2^{0} \times 2^{0} \times 2$$

Ce calcul nécessite donc un total de 10 multiplications.

Dans le calcul naïf, 2^{53} comporte, par définition, un total de 53 facteurs et son calcul direct nécessite donc un total de 52 multiplications …

Correction de l'algorithme

3. Une exécution de `fexpo(a,n)` se termine-t-elle toujours ?

Oui. Le second argument fourni à la fonction `fexpo` décroît strictement (à chaque appel, on fournit le quotient du précédent dans sa division euclidienne par 2). Ce second argument finit donc par valoir 0 et, dès lors, il n'y a plus d'appel à `fexpo`.

4. Démontrer que l'exécution de `fexpo(a,n)` retourne bien la valeur de a^n (on démontrera le résultat en raisonnant par récurrence sur *n*).

Nous allons ici mener une récurrence forte.

Nous considérons ici, pour tout entier naturel *n*, la propriété :

$\mathscr{P}_n$: « `fexpo(a,n)` retourne la valeur de a^n »

Initialisation

Pour `n=0`, l'appel `fexpo(a,n)`, d'après la définition de la fonction `fexpo`, retourne directement la valeur 1 qui est bien égale à a^0.

La propriété $\mathscr{P}_0$ est vraie.

Hérédité

Soit n un entier naturel quelconque fixé.
On suppose que, pour tout entier k inférieur ou égal à n, la propriété $\mathscr{P}_k$ est vraie et on s'intéresse à $\mathscr{P}_{n+1}$.

La division euclidienne de n par 2 s'écrit : $n = 2q + r$ avec $r = 0$ ou $r = 1$.

1er cas : $r = 0$ (n pair).

On a donc $n = 2q$ puis $n+1 = 2q+1$ qui est la division euclidienne de $n+1$ par 2 puisque $0 \leq 1 < 2$. L'appel `fexpo(a,n+1)` conduira donc à l'appel `fexpo(a,q)`.
D'après notre hypothèse de récurrence, cet appel renverra la valeur a^q et comme le reste de la division euclidienne de $n+1$ par 2 est égal à 1, le calcul complet effectué sera `prod * prod * a` et la valeur finalement renvoyée vaudra :

$$a^q \times a^q \times a = a^{2q+1} = a^{n+1}$$

2ème cas : $r = 1$ (n impair).

On a donc $n = 2q+1$ puis $n+1 = 2q+1+1 = 2\times(q+1)+0$ qui est la division euclidienne de $n+1$ par 2. L'appel `fexpo(a,n+1)` conduira donc à l'appel `fexpo(a,q+1)`.
D'après notre hypothèse de récurrence, cet appel renverra la valeur a^{q+1} et comme le reste de la division euclidienne de $n+1$ par 2 est égal à 0, le calcul complet effectué sera `prod * prod` et la valeur finalement renvoyée vaudra :

$$a^{q+1} \times a^{q+1} = a^{2q+2} = a^{n+1}$$

Ainsi, dans les deux cas, on obtient bien a^{n+1} et on en conclut que la propriété $\mathscr{P}_{n+1}$ est vraie.

Conclusion

Pour tout n entier naturel, `fexpo(a,n)` retourne la valeur de a^n.

Complexité de l'algorithme

Dans cette partie, on se donne un réel a non nul. On définit alors les suites $M = (M_n)_{n\in\mathbb{N}}$ et $C = (C_n)_{n\in\mathbb{N}}$ où, pour tout entier naturel n, les entiers M_n et A_n correspondent respectivement au nombre total de multiplications et au nombre total d'appels à la fonction `fexpo` engendrés par un premier appel à cette fonction : `fexpo(a,n)` (cet appel initial ne sera donc pas comptabilisé au niveau de la suite C). Ces quantités ne dépendent pas de la valeur du réel `a`.

Pour l'étude expérimentale des suites *M* et *C*, on écrira un programme mettant en œuvre la fonction `fexpo` et incluant des variables globales `m` et `c` donnant, à la fin de l'exécution, le nombre total de multiplications effectuées et le nombre total d'appels à la fonction `fexpo`.

Le programme `ExpoRapide` ci-dessous met en œuvre cette suggestion :

```
def fexpo(a,n):
    global m, c
    if n == 0:
        return 1
    else:
        q,r = divmod(n,2)
        c += 1
        prod = fexpo(a,q)
        prod = prod * prod
        m += 1
        if r == 1:
            prod = prod * a
            m += 1
        return prod

# PROGRAMME PRINCIPAL
# ===================

global m, c
m, c = 0, 0

# Saisie du réel a
# ----------------

a = float(input('Veuillez saisir le réel a : '))

# Saisie de l'entier n
# --------------------

n = int(input('\nVeuillez saisir l\'entier n : '))

# Calcul de fexpo(a,n)
# --------------------

e = fexpo(a,n)

# Affichage du résultat
# ---------------------

print('\n',a,' exposant ',n,' vaut : ',e)
print('\nNombre total de multiplications : ',m)
print('\nNombre total d\'appels à la fonction fexpo : ',c)

# FIN DU PROGRAMME PRINCIPAL
# ==========================
```

Etude de la suite *M*.

5. Effectuer une étude expérimentale de la monotonie de la suite *M*.

A l'aide du programme ci-dessus, on obtient facilement :

n	M_n
0	0
1	2
2	3
3	4
4	4
5	5
6	5
7	6
8	5

Si les premières valeurs de M_n (et notre intuition ?) peuvent nous laisser penser que la suite *M* est croissante, la valeur de M_8 invalide en partie ce « résultat » : si la suite M est croissante, elle ne l'est pas à partir de $n = 0$.

En poursuivant un peu, on obtient :

n	M_n
8	5
9	6
10	6
11	7
12	6
13	7
14	7
15	8
16	6

Les choses ne semblent pas vraiment s'éclaircir et il serait bien hasardeux de penser que la suite *M* puisse être croissante, ne serait-ce qu'à partir d'un certain rang …

6. Soit *n* un entier naturel non nul.
   On écrit $n = 2q + r$ la division euclidienne de *n* par 2 et on définit la suite $(u_n)_{n\in\mathbb{N}^*}$ par : $\forall n \in \mathbb{N}^*, M_n = M_q + u_n$.
   Préciser u_n en fonction de la parité de *n*.

On a, pour $n \neq 0$: $a^n = a^{2q+r} = a^q \times a^q \times a^r$.

Si n est pair, on a $r=0$ et $a^n = a^{2q} = a^q \times a^q$.
Ainsi, le nombre M_n de multiplication requis pour le calcul de a^n est égal au nombre de multiplications requis pour le calcul de a^q augmenté de 1 (soit la multiplication $a^q \times a^q$). Dans ce cas, on a donc : $M_n = M_q + 1$.

Si n est impair, on a : $r=1$ et $a^n = a^{2q+1} = a^q \times a^q \times a$.
Ainsi, le nombre M_n de multiplication requis pour le calcul de a^n est égal au nombre de multiplications requis pour le calcul de a^q augmenté de 2 (soit la multiplication $a^q \times a^q \times a$). Dans ce cas, on a donc : $M_n = M_q + 2$.

On peut synthétiser le résultat sous la forme d'une unique relation :

$$M_n = M_q + 1 + r = M_q + 1 + \frac{1+(-1)^{n+1}}{2}$$

7. Pour tout entier naturel p, calculer M_{2^p}, M_{2^p-1} et M_{2^p+1}.
Que peut-on en conclure sur la monotonie de M ?

En utilisant la question 6, il vient, pour tout entier naturel non nul p :

$$M_{2^p} = M_{2^{p-1}} + 1 = M_{2^{p-2}} + 2 = \ldots = M_{2^0} + p = M_1 + p$$

Or, un appel de « `fexpo` » avec $n=1$ engendre deux multiplications car $a^1 = a^0 \times a^0 \times a$. On a donc $M_1 = 2$ et, finalement : $M_{2^p} = p+2$.

Par ailleurs, pour tout entier p non nul, on a : $2^p - 1 = \left(2^p - 2\right) + 1 = 2\left(2^{p-1} - 1\right) + 1$.

Donc, toujours en s'appuyant sur les résultats de la question 6 :

$$M_{2^p-1} = M_{2^{p-1}-1} + 2 = M_{2^{p-2}-1} + 4 = M_{2^{p-3}-1} + 6 = \ldots = M_{2^0-1} + 2p = M_0 + 2p$$

Or, un appel de « `fexpo` » avec $n=0$ n'engendre pas de multiplication. On a donc $M_0 = 0$ et, finalement : $M_{2^p-1} = 2p$. On constate que cette égalité reste valable pour $p=0$.

Par exemple avec $p=6$, on a : $2^6 - 1 = 64 - 1 = 63$ et :

$$\begin{aligned}
a^{63} &= 2^{31} \times 2^{31} \times 2 \\
2^{31} &= 2^{15} \times 2^{15} \times 2 \\
2^{15} &= 2^{7} \times 2^{7} \times 2 \\
2^{7} &= 2^{3} \times 2^{3} \times 2 \\
2^{3} &= 2^{1} \times 2^{1} \times 2 \\
2^{1} &= 2^{0} \times 2^{0} \times 2
\end{aligned}$$

Soit un total de $12 = 2\times 6$ multiplications.

Enfin, pour tout entier p non nul, on a : $2^p + 1 = 2\times 2^{p-1} + 1$.
Donc : $M_{2^p+1} = M_{2^{p-1}} + 2 = \left(\left(p-1\right)+2\right)+2 = p+3$.

Finalement :

$$\boxed{\forall p \in \mathbb{N},\quad M_{2^p} = p+2,\ M_{2^p-1} = 2p \text{ et } M_{2^p+1} = p+3}$$

Pour tout entier naturel *p*, on a donc :

$$M_{2^p} - M_{2^p-1} = p+2-2p = 2-p$$
$$M_{2^p+1} - M_{2^p} = p+3-\left(p+2\right) = 1$$

Ainsi, pour tout entier naturel *p* strictement supérieur à 2, on a :

$$M_{2^p} - M_{2^p-1} = 2-p < 0 \text{ et } M_{2^p+1} - M_{2^p} = 1 > 0$$

En d'autres termes, on peut trouver un indice *n* arbitrairement grand (de la forme 2^p) tel que l'on ait : $M_n - M_{n-1} < 0$ et $M_{n+1} - M_n > 0$. On en conclut immédiatement que la suite *M* n'est pas monotone.

8. Soit *n* un entier naturel non nul.
Montrer qu'il existe un unique entier naturel *p* tel que : $2^p \le n < 2^{p+1}$.
Préciser *p*.

La fonction $x \mapsto 2^x$ est strictement croissante sur $\mathbb{R}_+$ et prend ses valeurs dans $\left[1\,;+\infty\right[$. Elle y est continue en tant que fonction exponentielle ($2^x = e^{x\times\ln 2}$).
Elle définit donc une bijection de $\mathbb{R}_+$ dans $\left[1\,;+\infty\right[$ et, pour tout *n* entier naturel non nul, il existe donc un unique réel x_n tel que $n = 2^{x_n}$. Ce réel est bien sûr égal au logarithme népérien de base 2 de *n* : $x_n = \log_2\left(n\right)$.
On a alors les équivalences :

$$2^p \le n < 2^{p+1} \Leftrightarrow 2^p \le 2^{\log_2(n)} < 2^{p+1} \Leftrightarrow p \le \log_2\left(n\right) < p+1$$

En d'autres termes : $p = \mathrm{E}\left(\log_2\left(n\right)\right)$.

9. A l'aide d'une récurrence forte, montrer alors que l'on a : $p+2 \leq M_n \leq 2(p+1)$.

D'après la question précédente, il convient donc de montrer que l'on a, pour tout *n* entier naturel non nul :

$$\mathrm{E}(\log_2(n))+2 \leq M_n \leq 2(\mathrm{E}(\log_2(n))+1)$$

Pour $n=1$, on a : $\mathrm{E}(\log_2(n))=\mathrm{E}(\log_2(1))=\mathrm{E}(0)=0$.

Donc : $2(\mathrm{E}(\log_2(n))+1)=2(\mathrm{E}(\log_2(1))+1)=2\times(0+1)=2$.

Or $M_1=1$. Ainsi, on a bien : $\mathrm{E}(\log_2(1))+2 \leq M_1 \leq 2(\mathrm{E}(\log_2(1))+1)$.

La propriété est initialisée.

Soit maintenant un entier naturel *n* non nul quelconque fixé.
On suppose que la propriété est vraie à tous les rangs jusqu'au rang *n* (compris).

D'après la question précédente, nous avons $2^{\mathrm{E}(\log_2(n+1))} \leq n+1 < 2^{\mathrm{E}(\log_2(n+1))+1}$ et nous sommes ainsi amenés à distinguer plusieurs cas.

$\boxed{1^{\text{er}} \text{ cas : } n+1=2^{\mathrm{E}(\log_2(n+1))}}$

Dans ce cas, $n+1$ est simplement une puissance de 2 …
D'après la question 7, il vient immédiatement : $M_{n+1}=M_{2^{\mathrm{E}(\log_2(n+1))}}=\mathrm{E}(\log_2(n+1))+2$.

$\boxed{2^{\text{ème}} \text{ cas : } n+1=2^{\mathrm{E}(\log_2(n+1))+1}-1}$

D'après la question 7, il vient immédiatement :

$$M_{n+1}=M_{2^{\mathrm{E}(\log_2(n+1))+1}-1}=2(\mathrm{E}(\log_2(n+1))+1).$$

$\boxed{3^{\text{ème}} \text{ cas : } 2^{\mathrm{E}(\log_2(n+1))} < n+1 < 2^{\mathrm{E}(\log_2(n+1))+1}-1}$

C'est le cas général.
On pose : $n+1=2q+r$ avec $r=0$ ou 1.
D'après la question 6 on a alors : $M_{n+1}=M_q+1+r$.
D'après l'hypothèse de récurrence, on a : $\mathrm{E}(\log_2(q))+2 \leq M_q \leq 2(\mathrm{E}(\log_2(q))+1)$.

- Si $r=0$
  On a donc $n+1=2q$ et $M_{n+1}=M_q+1$.
  D'où : $\log_2(n+1)=\log_2(2q)=\log_2(2)+\log_2(q)=1+\log_2(q)$.
  Alors : $\mathrm{E}(\log_2(n+1))=\mathrm{E}(1+\log_2(q))=1+\mathrm{E}(\log_2(q))$.

De la double inégalité $\mathrm{E}\left(\log_2(q)\right)+2 \leq M_q \leq 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)$, on tire alors :

$$\mathrm{E}\left(\log_2(q)\right)+2+1 \leq M_q+1 \leq 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)+1$$

Soit : $\mathrm{E}\left(\log_2(n+1)\right)+2 \leq M_{n+1} \leq 2\,\mathrm{E}\left(\log_2(n+1)\right)+1$.

Comme $2\,\mathrm{E}\left(\log_2(n+1)\right)+1<2\,\mathrm{E}\left(\log_2(n+1)\right)+2=2\left(\mathrm{E}\left(\log_2(n+1)\right)+1\right)$, il vient finalement :

$$\mathrm{E}\left(\log_2(n+1)\right)+2 \leq M_{n+1} \leq 2\left(\mathrm{E}\left(\log_2(n+1)\right)+2\right)$$

C'est l'encadrement recherché.

- Si $r=1$

  On a donc $n+1=2q+1$ et $M_{n+1}=M_q+2$.

  D'où : $\log_2(n+1)=\log_2(2q+1)=\log_2(2)+\log_2\left(q+\frac{1}{2}\right)=1+\log_2\left(q+\frac{1}{2}\right)$.

  Alors : $\mathrm{E}\left(\log_2(n+1)\right)=\mathrm{E}\left(1+\log_2\left(q+\frac{1}{2}\right)\right)=1+\mathrm{E}\left(\log_2\left(q+\frac{1}{2}\right)\right)$.

  Comme on a : $q \geq 1$ ($q=0$ entraîne $n=0$ ici …), il vient : $q<q+\frac{1}{2}<2q$ et donc : $\log_2(q)<\log_2\left(q+\frac{1}{2}\right)<\log_2(2q)$ puis enfin :

$$\mathrm{E}\left(\log_2(q)\right) \leq \mathrm{E}\left(\log_2\left(q+\frac{1}{2}\right)\right) \leq \mathrm{E}\left(\log_2(2q)\right)=1+\mathrm{E}\left(\log_2(q)\right) \qquad (1)$$

  De la double inégalité $\mathrm{E}\left(\log_2(q)\right)+2 \leq M_q \leq 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)$, on tire alors :

$$\mathrm{E}\left(\log_2(q)\right)+2+2 \leq M_q+2 \leq 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)+2$$

  c'est-à-dire : $\mathrm{E}\left(\log_2(q)\right)+2+2 \leq M_{n+1} \leq 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)+2$.

  De (1) on tire :

$$\begin{aligned}\mathrm{E}\left(\log_2(q)\right)+2+2 &= \mathrm{E}\left(\log_2(q)\right)+1+1+2\\ &\geq \mathrm{E}\left(\log_2\left(q+\frac{1}{2}\right)\right)+1+2=\mathrm{E}\left(\log_2(n+1)\right)+2\end{aligned}$$

  Et :

$$\begin{aligned}2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)+2 &\leq 2\left(\mathrm{E}\left(\log_2\left(q+\frac{1}{2}\right)\right)+1\right)+2\\ &=2\,\mathrm{E}\left(\log_2(n+1)\right)+2=2\left(\mathrm{E}\left(\log_2(n+1)\right)+1\right)\end{aligned}$$

Finalement :

$$\mathrm{E}\left(\log_2(q)\right)+2+2 \le M_{n+1} \le 2\left(\mathrm{E}\left(\log_2(q)\right)+1\right)+2$$
$$\Rightarrow \mathrm{E}\left(\log_2(n+1)\right)+2 \le M_{n+1} \le 2\left(\mathrm{E}\left(\log_2(n+1)\right)+1\right)$$

Ainsi, la propriété est vraie au rang $n+1$, elle est héréditaire.

La propriété est ainsi vraie pour tout entier naturel *n* non nul :

$$\boxed{\forall n \in \mathbb{N}^*,\ \mathrm{E}\left(\log_2(n)\right)+2 \le M_n \le 2\left(\mathrm{E}\left(\log_2(n)\right)+1\right)}$$

10. Déduire de la question précédente que l'on a : $M_n = \mathrm{O}(\ln n)$.

A la question précédente, on a montré que l'on avait :

$$\mathrm{E}\left(\log_2(n)\right)+2 \le M_n \le 2\left(\mathrm{E}\left(\log_2(n)\right)+1\right)$$

De cette double inégalité, on tire immédiatement : $M_n = \mathrm{O}\left(\mathrm{E}\left(\log_2(n)\right)\right)$.
Mais comme $\mathrm{E}\left(\log_2(n)\right) \underset{+\infty}{\sim} \log_2(n)$, il vient : $M_n = \mathrm{O}\left(\log_2(n)\right)$.
Enfin, comme le logarithme de base 2 est proportionnel au logarithme népérien ($\log_2(n) = \dfrac{\ln n}{\ln 2}$), il vient : $M_n = \mathrm{O}\left(\ln(n)\right)$.

$$\boxed{M_n = \mathrm{O}(\ln n)}$$

Etude de la suite *A*.

11. Effectuer une étude expérimentale de la monotonie de la suite *A*.

A l'aide du programme utilisé dans la première question, on obtient facilement :

n	A_n
0	0
1	1
2	2
3	2
4	3
5	3
6	3
7	3
8	4
9	4
10	4
11	4
12	4
13	4
14	4
15	4
16	5

Il semble donc que la suite *A* soit croissante …Plus précisément, il semble que *A* prenne la valeur $p+1$ sur tout intervalle de la forme $[\![2^p\,;2^p-1]\!]$.

12. Pour *n* entier naturel non nul, on note $n=2q+r$ la division euclidienne de *n* par 2. Après avoir remarqué que l'on a $A_n=A_q+1$, montrer, à l'aide d'une récurrence forte, que la suite *A* est croissante.

Pour *n* entier naturel non nul, l'appel `fexpo(a,n)` engendre l'appel `fexpo(a,q)` où `q` est le reste de la division euclidienne de *n* par 2. On a donc immédiatement $A_n=A_q+1$.

Posons, pour tout *n* entier naturel : $\mathscr{P}_n$: « $A_{n+1}-A_n\geq 0$ ».

<u>Initialisation</u>

On a $A_0=0$ et $A_1=1$.
On a donc $A_1-A_0=1-0=1\geq 0$. $\mathscr{P}_0$ est donc vraie.

Hérédité

Soit n un entier naturel non nul quelconque fixé.
On suppose que pour tout entier k non nul inférieur ou égal à n, la propriété $\mathscr{P}_k$ est vraie.
On s'intéresse à $\mathscr{P}_{n+1}$, c'est-à-dire à la différence $A_{n+2} - A_{n+1}$.
On pose $n+1 = 2q + r$, la division euclidienne de $n+1$ par 2.

$\boxed{1^{\text{er}} \text{ cas : } r = 0}$

D'après la remarque préliminaire, on a : $A_{n+1} = A_q + 1$.
Par ailleurs $n+1 = 2q \Leftrightarrow n+2 = 2q+1$ est la division euclidienne de $n+2$ par 2. On a donc : $A_{n+2} = A_q + 1$, puis : $A_{n+2} - A_{n+1} = A_q + 1 - \left(A_q + 1\right) = 0$.

$\boxed{2^{\text{ème}} \text{ cas : } r = 1}$

On a encore $A_{n+1} = A_q + 1$ mais cette fois $n+1 = 2q+1 \Leftrightarrow n+2 = 2q+2 = 2\left(q+1\right)+0$.
D'où : $A_{n+2} = A_{q+1} + 1$, puis : $A_{n+2} - A_{n+1} = A_{q+1} + 1 - \left(A_q + 1\right) = A_{q+1} - A_q$.
L'hypothèse de récurrence nous donne alors $A_{q+1} - A_q \geq 0$ et on en déduit $A_{n+2} - A_{n+1} \geq 0$.

Dans tous les cas, on a bien : $A_{n+2} - A_{n+1} \geq 0$. La suite A est croissante.

13. Pour tout entier naturel p, calculer A_{2^p}.

Pour tout entier naturel p supérieur ou égal à 1, on a : $2^p = 2 \times 2^{p-1} + 0$. En d'autres termes, le quotient de la division euclidienne de 2^p par 2 est égal à 2^{p-1}.
D'après la question précédente, on a donc : $A_{2^p} = A_{2^{p-1}} + 1$. Il vient alors (récurrence immédiate) : $A_{2^p} = A_{2^0} + p = A_1 + p$. Comme $1 = 2 \times 0 + 1$, on a immédiatement $A_1 = 1$ et, finalement : $A_{2^p} = p + 1$. Cette formule reste valable pour $p = 0$.

$$\boxed{\forall p \in \mathbb{N},\ A_{2^p} = p + 1}$$

14. A l'aide de la question 8, déterminer un encadrement puis un équivalent de A_n.

A la question 8, on a montré que l'on avait, pour tout entier naturel n non nul :

$$2^{\mathrm{E}\left(\log_2(n)\right)} \leq n < 2^{\mathrm{E}\left(\log_2(n)\right)+1}$$

La suite A étant croissante, il en découle :

$$A_{2^{\mathrm{E}(\log_2(n))}} \leq A_n \leq A_{2^{\mathrm{E}(\log_2(n))+1}}$$

Soit, en tenant compte du résultat de la question précédente :

$$\mathrm{E}\left(\log_2(n)\right)+1 \leq A_n \leq \mathrm{E}\left(\log_2(n)\right)+1+1$$

C'est-à-dire :

$$\mathrm{E}\left(\log_2(n)\right)+1 \leq A_n \leq \mathrm{E}\left(\log_2(n)\right)+2$$

Pour tout entier naturel *n* supérieur ou égal à 2, il vient alors :

$$\frac{\mathrm{E}\left(\log_2(n)\right)+1}{\mathrm{E}\left(\log_2(n)\right)} \leq \frac{A_n}{\mathrm{E}\left(\log_2(n)\right)} \leq \frac{\mathrm{E}\left(\log_2(n)\right)+2}{\mathrm{E}\left(\log_2(n)\right)}$$

C'est-à-dire :

$$1+\frac{1}{\mathrm{E}\left(\log_2(n)\right)} \leq \frac{A_n}{\mathrm{E}\left(\log_2(n)\right)} \leq 1+\frac{2}{\mathrm{E}\left(\log_2(n)\right)}$$

Comme $\lim\limits_{n\to+\infty} \mathrm{E}\left(\log_2(n)\right)=+\infty$, on a $\lim\limits_{n\to+\infty} \frac{1}{\mathrm{E}\left(\log_2(n)\right)}=0$ et le théorème des gendarmes nous permet de conclure : $\lim\limits_{n\to+\infty} \frac{A_n}{\mathrm{E}\left(\log_2(n)\right)}=1$, soit :

$$A_n \underset{+\infty}{\sim} \mathrm{E}\left(\log_2(n)\right)$$

Or : $\mathrm{E}\left(\log_2(n)\right) \underset{+\infty}{\sim} \log_2(n)$. Donc, finalement :

$$\boxed{A_n \underset{+\infty}{\sim} \log_2(n)}$$