
TP2 - Question 1

réalisé par Claire Bouttes - 536793110
IFT-3001

Table des matières

Introduction	2
1 Analyse théorique	2
1.1 Algorithme 1	2
1.2 Algorithme 2	3
2 Analyse empirique	4
2.1 Algorithme 1	4
2.2 Algorithme 2	6
Conclusion	9

Introduction

Pour cette question, il sera étudié l'efficacité de deux algorithmes probabilistes premièrement théoriquement puis empiriquement.

Les deux algorithmes ayant des similitudes, nous pouvons identifier une **opération de base** qui sera commune aux deux analyses. Cette dernière sera $c \leftarrow c + 1$.

De plus, $A[0]$ étant analysé dans les deux cas, le temps d'exécution ne dépend pas uniquement de la taille de l'instance mais aussi de l'instance elle-même. Nous aurons donc un **pire cas**, un **meilleur cas** et un **cas moyen**.

On désignera par ρ , la collection des nombres aléatoires utilisés par les deux algorithmes au cours de son exécution sur n .

1 Analyse théorique

1.1 Algorithme 1

Algorithm 1: AlgorithmeProbabiliste1($A[0..n-1]$)

```
 $c \leftarrow 0;$ 
 $k \leftarrow UNIFORME(0, n - 1);$ 
if  $A[0] = k$  then
    for  $i = k..n$  do
        for  $j = 1..k$  do
             $c \leftarrow c + 1;$ 
        end
    end
end
return  $c;$ 
```

Pire cas $C_{worst}(n) = E_{\rho}C_{worst}(n, \rho) = \sum_{\rho} p(\rho)C_{worst}(n, \rho)$

$$C_{worst}(n) = \sum_{\rho=0}^{n-1} p(\rho)C_{worst}(n, \rho)$$

$$C_{worst}(n) = \sum_{\rho=0}^1 p(\rho)C_{worst}(n, \rho) + \sum_{i=2}^{n-1} p(\rho)C_{worst}(n, \rho)$$

Dans le pire cas, $A[0]$ est égal à 0.

$$C_{worst}(n) = \sum_{\rho=0}^1 p(\rho) \sum_{i=\rho}^n \sum_{j=1}^{\rho} 1 + \sum_{i=2}^{n-1} p(\rho)0$$

$$C_{worst}(n) = \sum_{\rho=0}^1 p(\rho) \sum_{i=\rho}^1 (\rho - 1 + 1) \times 1$$

$$C_{worst}(n) = \sum_{\rho=0}^1 p(\rho)\rho(n - \rho + 1)$$

$$C_{worst}(n) = \sum_{\rho=0}^1 p(\rho)(-\rho^2 + n\rho + 1)$$

$$C_{worst}(n) = p(0)(-0^2 + n * 0 + 1) + p(1)(-1^2 + n \times 1 + 1)$$

$$C_{worst}(n) = p(0) + p(1)(n + 1)$$

$p(0)$ et $p(1)$ étant des constantes de probabilités entre 0 et 1. On pourrait en déduire que l'algorithme est en $\theta(n)$:

$$p(\{0, 1\}) + np(1) > p(1)n$$

$$p(\{0, 1\}) + np(1) \leq n + n$$

Donc on peut en conclure que en pire cas, l'algorithme est en $\theta(n)$.

Meilleur cas Dans le meilleur cas, on va reprendre le début de raisonnement du pire cas, sauf que cette fois-ci, $A[0]$ est toujours différent de k . Nous avons alors :

$$C_{best}(n) = E_{\rho}C_{best}(n, \rho) = \sum_p p(\rho)C_{best}(n, \rho) = 0$$

Cas moyen En cas moyen, il y a une chance sur deux comme indiqué dans l'énoncé que $A[0]$ soit 1 ou 0.

$$\begin{aligned} C_{avg}(n) &= \sum_p p(\rho)C_{avg}(n, \rho) \\ C_{avg}(n) &= \sum_p p(\rho)\frac{1}{2}(\sum_{i=\rho}^n \sum_{j=1}^{\rho} 1) + \frac{1}{2}(0) \\ C_{avg}(n) &= \frac{1}{2} \sum_p p(\rho) \sum_{i=\rho}^n \sum_{j=1}^{\rho} 1 \\ C_{avg}(n) &= \frac{1}{2}(p(0) + p(1)(n + 1)) \end{aligned}$$

Ainsi l'algorithme serait en $\theta(n)$.

1.2 Algorithme 2

Algorithm 2: AlgorithmeProbabiliste2($A[0..n-1]$)

```

c ← 0;
k ← UNIFORME(0, n - 1);
if A[0] = 0 then
    for i = 1..k do
        | c ← c + 1;
    end
end
return c;
```

Pire cas $C_{worst}(n) = E_{\rho}C_{worst}(n, \rho) = \sum_p p(\rho)C_{worst}(n, \rho)$

$$C_{worst}(n) = \sum_{\rho=0}^{n-1} p(\rho)C_{worst}(n, \rho)$$

Dans le pire cas, $A[0]$ est toujours égal à 0.

$$C_{worst}(n) = \sum_{\rho=0}^{n-1} p(\rho) \sum_{i=0}^{\rho} 1$$

$$C_{worst}(n) = \sum_{\rho=0}^{n-1} p(\rho)(\rho - 1 + 1) \times 1$$

$$C_{worst}(n) = \sum_{\rho=0}^{n-1} p(\rho)\rho$$

Il n'est pas possible de donner une valeur de θ sans valeurs de ρ .

Meilleur Cas $C_{best}(n) = E_{\rho}C_{best}(n, \rho) = \sum_p p(\rho)C_{best}(n, \rho)$

Dans le meilleur cas, $A[0]$ est toujours différent de 0. Nous avons alors peut importe la valeur de k : $C_{best}(n) = 0$.

Cas moyen $C_{avg}(n) = E_{\rho}C_{avg}(n, \rho) = \sum_p p(\rho)C_{avg}(n, \rho)$

$$C_{avg}(n) = \sum_{\rho=0}^{n-1} p(\rho)C_{avg}(n, \rho)$$

En cas moyen, de la même manière que pour le premier algorithme, il y a une chance sur deux que $A[0]$ soit 1 ou 0.

$$C_{avg}(n) = \sum_{\rho=0}^{n-1} p(\rho) \left(\frac{1}{2}(0) + \frac{1}{2}(\rho) \right)$$

$$C_{avg}(n) = \frac{1}{2} \sum_{\rho=0}^{n-1} p(\rho) \rho$$

Comme pour le pire cas, il n'est pas possible de donner pour le moment une valeur de θ sans connaître la valeur de ρ .

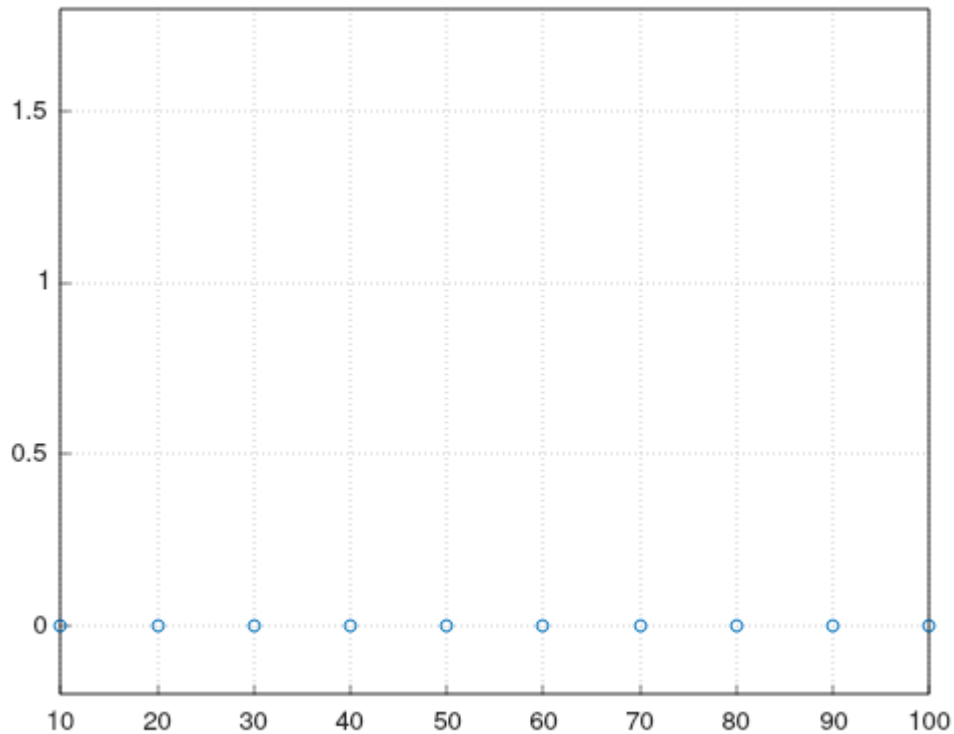
2 Analyse empirique

Le résultat des graphiques correspond au nombre d'exécution de l'opération baromètre en fonction de n , la taille du tableau de bits en entrée.

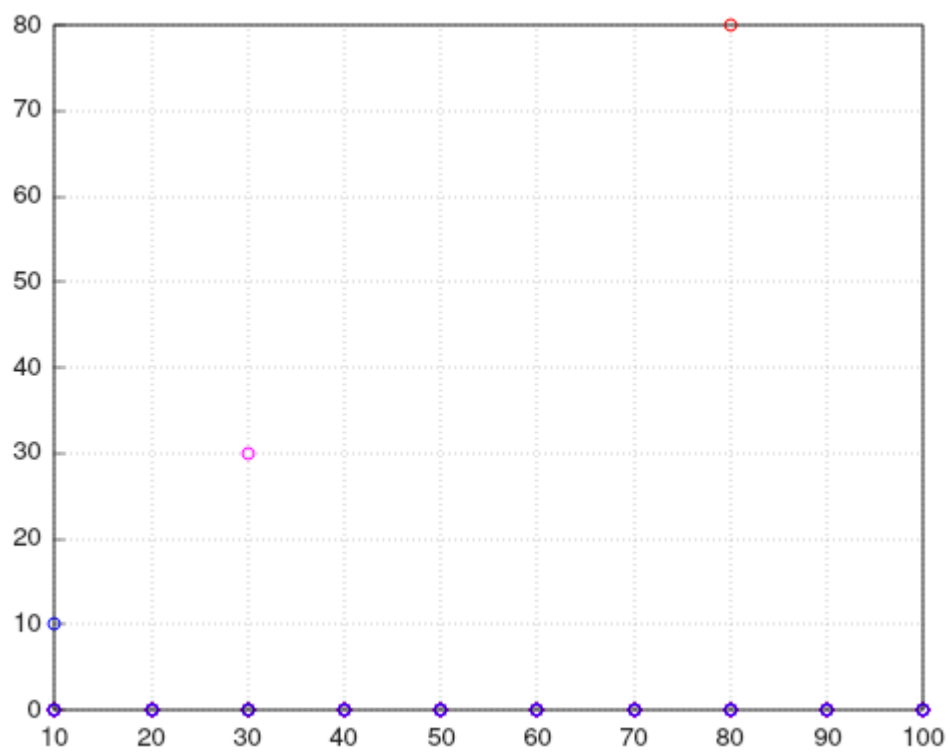
2.1 Algorithme 1

Pour le pire cas, $A[0]$ est égal à true (1). Pour le meilleur cas, $A[0]$ est égal à false (0).

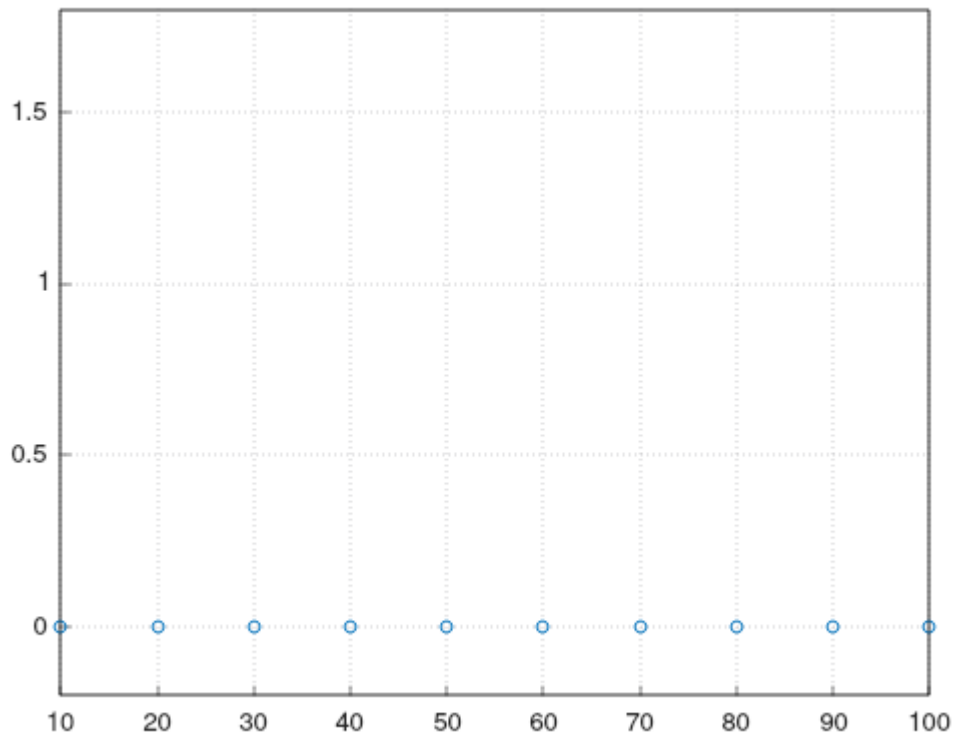
Cas moyen



Meilleur Cas



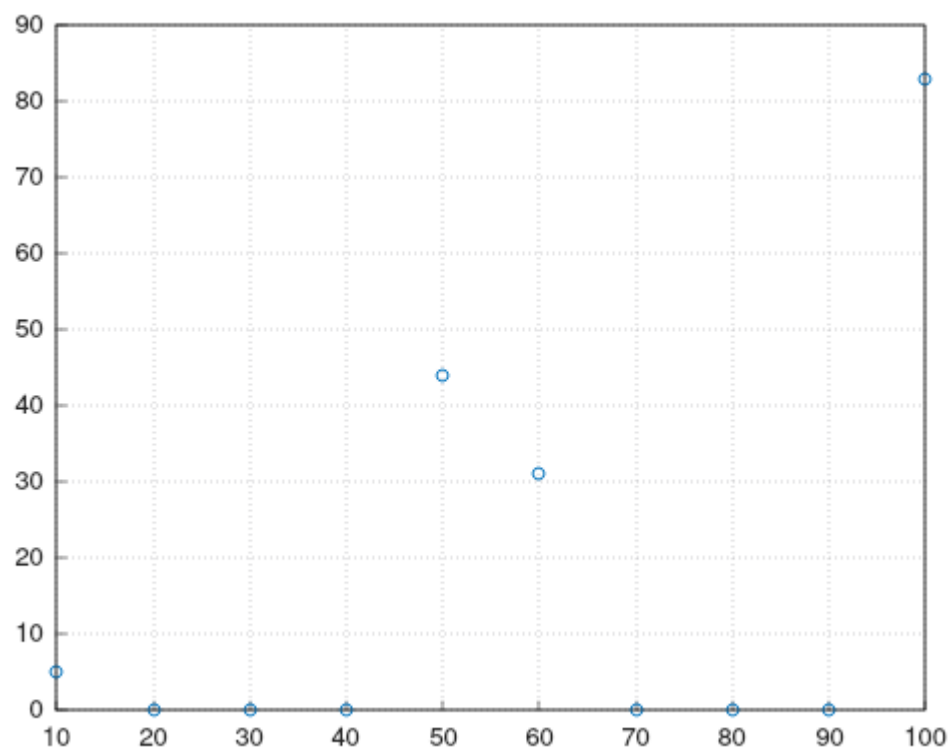
Pire Cas



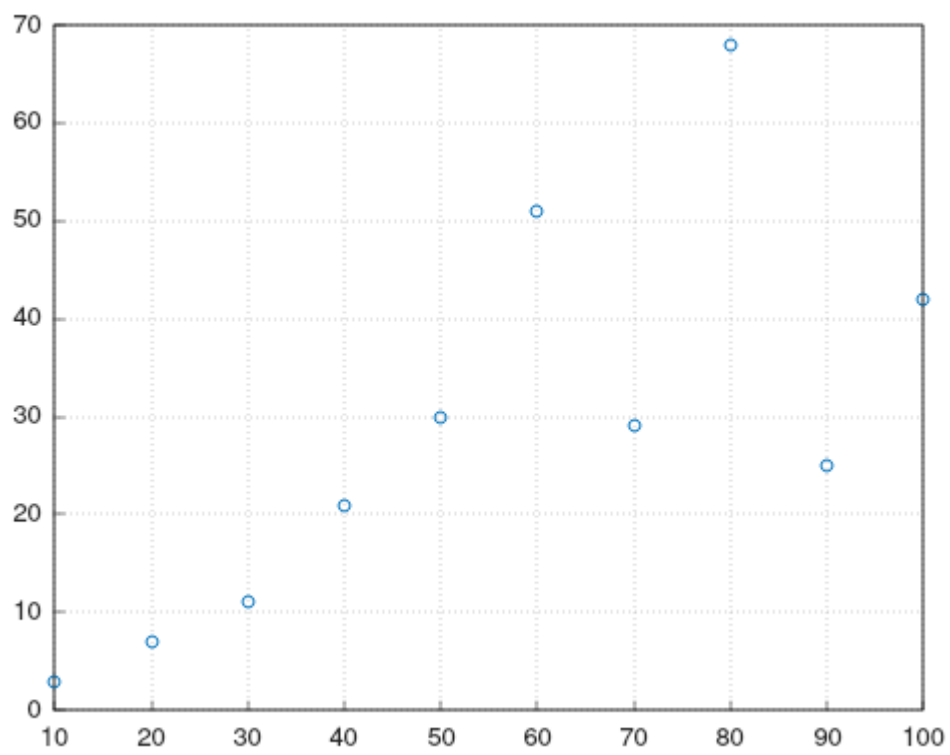
2.2 Algorithme 2

Pour le pire cas, $A[0]$ est égal à false (0). Pour le meilleur cas, $A[0]$ est égal à true (1).

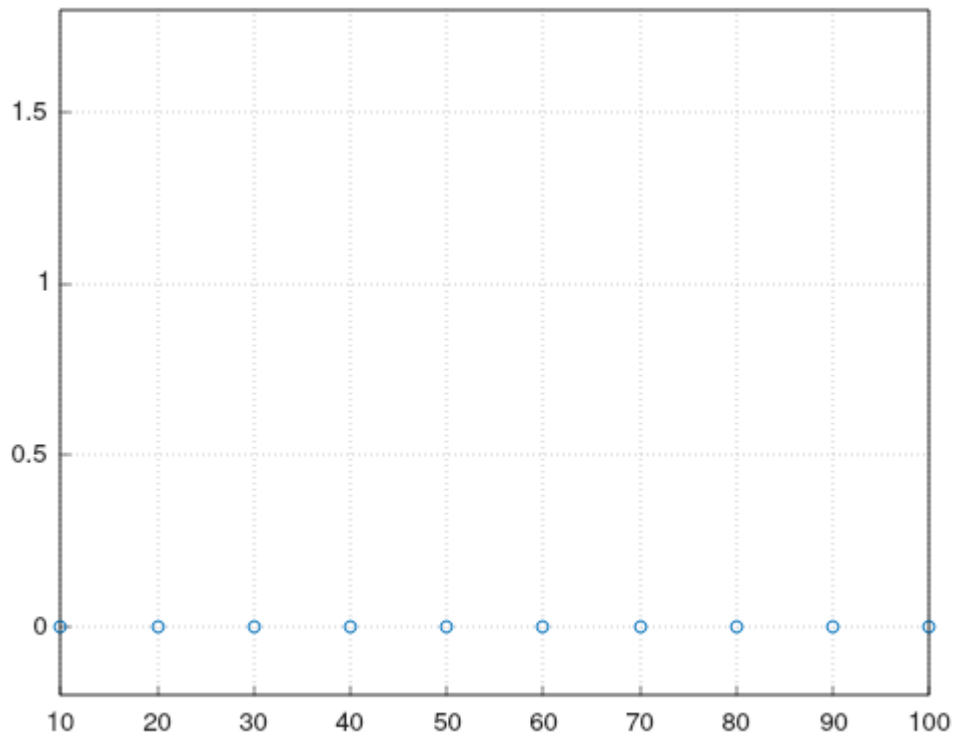
Cas moyen



Meilleur Cas



Pire Cas



Conclusion

En conclusion, il faudrait bien plus de 10 essais pour vraiment sortir des probabilités de sortie. De plus, j'ai travaillé avec une librairie c++ pour pouvoir afficher les graphiques. Malheureusement, de temps en temps la librairie n'affichait pas correctement les graphiques.