



**PRÉPASIMPLE**

prepasimple.fr

---

**ESSEC**

**Mathématiques 2**

**1990**

*Corrigé de mathématiques approfondies*

---

**Mathématiques approfondies — Prépa ECG**

*Annales de concours · Corrigé*

## Préliminaires

On pose

$$H_n = \sum_{j=1}^n \frac{1}{j}$$

pour  $n \geq 1$ . Ainsi  $u_n = H_n - \ln n$ ,  $v_n = H_n - \ln n - \frac{1}{n}$  et  $m_n = H_n - \ln n - \frac{1}{2n}$ .

### 1. Étude de deux suites adjacentes.

(a) Encadrement de l'accroissement du logarithme.

#### Méthode 1 — Intégration d'un encadrement.

Pour tout  $t \in [n, n+1]$ , avec  $n \geq 1$ , on a  $\frac{1}{n+1} \leq \frac{1}{t} \leq \frac{1}{n}$ . Les trois fonctions sont continues sur le segment. Par croissance de l'intégrale :

$$\frac{1}{n+1} \leq \int_n^{n+1} \frac{dt}{t} \leq \frac{1}{n}.$$

La primitive du terme central est  $\ln t$ . On obtient donc :

$$\frac{1}{n+1} \leq \ln(n+1) - \ln n \leq \frac{1}{n}.$$

#### Méthode 2 — Théorème des accroissements finis.

La fonction  $\ln$  est continue sur  $[n, n+1]$  et dérivable sur  $]n, n+1[$ . Le théorème des accroissements finis donne un réel  $c \in ]n, n+1[$  tel que :

$$\ln(n+1) - \ln n = \frac{1}{c}.$$

Comme  $n < c < n+1$ , on a même  $\frac{1}{n+1} < \frac{1}{c} < \frac{1}{n}$ , ce qui implique l'encadrement

demandé. L'intégration compare des aires, tandis que les accroissements finis comparent l'accroissement à une pente intermédiaire.

(b) Monotonie et caractère adjacent.

Calculons les différences successives :

$$u_{n+1} - u_n = \frac{1}{n+1} - (\ln(n+1) - \ln n) \leq 0,$$

$$v_{n+1} - v_n = u_{n+1} - u_n - \frac{1}{n+1} + \frac{1}{n} = \frac{1}{n} - (\ln(n+1) - \ln n) \geq 0.$$

Les inégalités proviennent de a). Ainsi  $(u_n)$  est décroissante et  $(v_n)$  croissante. De plus :

$$u_n - v_n = \frac{1}{n} \rightarrow 0.$$

Les deux suites sont donc adjacentes. Pour préciser leur convergence, on a  $v_1 = 0$ ,  $u_1 = 1$ , et  $v_n \leq u_n$  pour tout  $n$ . Ainsi  $0 \leq v_n \leq u_n \leq 1$ . Les deux suites sont monotones et bornées, donc elles convergent ; leur différence tendant vers 0, leurs limites sont égales.

En notant  $\gamma$  cette limite commune, la monotonie donne pour tout  $n \geq 1$  :

$$v_n \leq \gamma \leq u_n.$$

Question adaptée au programme actuel : la borne  $n \geq 1$  de la définition de  $u_n$ , dont le membre droit est incomplet dans le PDF, est explicitée conformément à la définition de  $v_n$  et aux termes  $1/n$  et  $\ln n$ .

## 2. Première approximation de la limite.

(a) Erreur de la moyenne des deux suites.

L'encadrement  $v_n \leq \gamma \leq u_n$  établi en 1.(b), après soustraction de  $m_n = (u_n + v_n)/2$ , donne :

$$-\frac{u_n - v_n}{2} \leq \gamma - m_n \leq \frac{u_n - v_n}{2}.$$

Comme  $u_n - v_n = 1/n$ , on en déduit :

$$|m_n - \gamma| \leq \frac{1}{2n}.$$

Cette borne vaut pour tout entier  $n \geq 1$  et garantit une approximation sans utiliser la valeur numérique de  $\gamma$ .

(b) Calcul de la moyenne en Python et approximations.

Question adaptée au programme actuel. Code adapté en Python.

La formule  $m_n = H_n - \ln n - 1/(2n)$  permet un calcul direct. La fonction suivante prend comme argument un entier  $n \geq 1$  et renvoie  $m_n$  :

### Python

```
import numpy as np

def moyenne(n):
    somme = 0.0
    for j in range(1, n + 1):
        somme = somme + 1.0 / j
    return somme - np.log(n) - 1.0 / (2 * n)
```

La variable `somme` contient  $H_n$  à la fin de la boucle. La borne finale de `range` est exclusive : `range(1, n + 1)` parcourt donc exactement les entiers de 1 à  $n$ . La commande `np.log` calcule le logarithme népérien. Le code a été exécuté pour les deux valeurs demandées.

On obtient :

$$m_5 = \frac{137}{60} - \ln 5 - \frac{1}{10} = \frac{131}{60} - \ln 5 \simeq 0,573895420899,$$

$$m_{50} = \sum_{j=1}^{50} \frac{1}{j} - \ln 50 - \frac{1}{100} \simeq 0,577182332901.$$

La question a) fournit les garanties :

$$|m_5 - \gamma| \leq 0,1, \quad |m_{50} - \gamma| \leq 0,01.$$

Les nombres  $m_5$  et  $m_{50}$  constituent donc respectivement les valeurs approchées demandées à 0,1 et 0,01 près. Les décimales affichées évaluent ces deux nombres ; la précision garantie pour la limite est celle des bornes d'erreur ci-dessus.

Pour juger a posteriori la qualité de  $m_5$  sans supposer  $\gamma$  connu, utilisons l'inégalité triangulaire :

$$|m_5 - \gamma| \leq |m_5 - m_{50}| + |m_{50} - \gamma| \leq |m_5 - m_{50}| + 0,01.$$

Or  $|m_5 - m_{50}| \simeq 0,003286912002$ . Ainsi :

$$|m_5 - \gamma| < 0,013287.$$

L'approximation obtenue avec seulement cinq termes est donc bien meilleure que la garantie initiale 0,1. La question suivante donnera une borne plus précise.

### 3. Amélioration de l'encadrement.

(a) Identité avec une intégrale.

Pour  $k \geq 1$ , calculons d'abord l'accroissement de  $m_k$  :

$$m_{k+1} - m_k = \frac{1}{k+1} - \ln\left(\frac{k+1}{k}\right) - \frac{1}{2(k+1)} + \frac{1}{2k},$$

soit :

$$m_{k+1} - m_k = \frac{1}{2k} + \frac{1}{2(k+1)} - \ln\left(\frac{k+1}{k}\right).$$

#### Méthode 1 — Développement du numérateur et calcul d'une primitive.

---

Posons  $J_k = \int_k^{k+1} \frac{(t-k)(k+1-t)}{t^3} dt$ . Le dénominateur ne s'annule pas sur le segment puisque  $k \geq 1$  ; l'intégrande est continue et l'intégrale existe. On développe :

$$(t-k)(k+1-t) = -t^2 + (2k+1)t - k(k+1).$$

Par conséquent :

$$J_k = \int_k^{k+1} \left( -\frac{1}{t} + \frac{2k+1}{t^2} - \frac{k(k+1)}{t^3} \right) dt.$$

Une primitive sur ce segment est :

$$-\ln t - \frac{2k+1}{t} + \frac{k(k+1)}{2t^2}.$$

L'évaluation aux bornes donne :

$$J_k = -\ln \left( \frac{k+1}{k} \right) + (2k+1) \left( \frac{1}{k} - \frac{1}{k+1} \right) + \frac{k(k+1)}{2} \left( \frac{1}{(k+1)^2} - \frac{1}{k^2} \right).$$

Les deux derniers termes valent respectivement  $(2k+1)/(k(k+1))$  et

$-(2k+1)/(2k(k+1))$ . Ainsi :

$$J_k = -\ln \left( \frac{k+1}{k} \right) + \frac{2k+1}{2k(k+1)} = m_{k+1} - m_k.$$

## Méthode 2 — Deux intégrations par parties sur un segment.

Cette méthode explique l'apparition du produit  $(t-k)(k+1-t)$ . Posons  $f(t) = 1/t$  et  $w(t) = (t-k)(k+1-t)$ . Les fonctions utilisées sont de classe  $\mathcal{C}^2$  sur le segment, et  $f''(t) = 2/t^3$ . De plus  $w(k) = w(k+1) = 0$ ,  $w'(k) = 1$ ,  $w'(k+1) = -1$ , et  $w''(t) = -2$ .

Une première intégration par parties donne :

$$J_k = \frac{1}{2} \int_k^{k+1} w(t) f''(t) dt = -\frac{1}{2} \int_k^{k+1} w'(t) f'(t) dt.$$

Une seconde donne :

$$J_k = -\frac{1}{2} [w'(t) f(t)]_k^{k+1} + \frac{1}{2} \int_k^{k+1} w''(t) f(t) dt,$$

d'où :

$$J_k = \frac{1}{2} \left( \frac{1}{k} + \frac{1}{k+1} \right) - \int_k^{k+1} \frac{dt}{t} = m_{k+1} - m_k.$$

Les deux méthodes établissent donc l'égalité exacte :

$$m_{k+1} - m_k = \int_k^{k+1} \frac{(t-k)(k+1-t)}{t^3} dt.$$

Le calcul de primitive est direct ; les intégrations par parties relient cette identité à la différence entre la moyenne des valeurs aux extrémités et l'intégrale de  $1/t$ .

(b) Sommation de l'encadrement et intervalle final.

Pour  $t \in [k, k+1]$ , posons  $s = t - k \in [0, 1]$ . Alors :

$$(t-k)(k+1-t) = s(1-s) = \frac{1}{4} - \left(s - \frac{1}{2}\right)^2.$$

Ainsi  $0 \leq (t-k)(k+1-t) \leq 1/4$ . Comme  $t^3 > 0$ , l'identité de a) et la croissance de l'intégrale donnent :

$$0 \leq m_{k+1} - m_k \leq \int_k^{k+1} \frac{dt}{4t^3} = \frac{1}{8} \left( \frac{1}{k^2} - \frac{1}{(k+1)^2} \right).$$

Pour des entiers  $n \geq 1$  et  $p \geq 1$ , additionnons ces inégalités pour  $k = n, n+1, \dots, n+p-1$ . Les deux sommes se télescopent :

$$0 \leq m_{n+p} - m_n \leq \frac{1}{8} \left( \frac{1}{n^2} - \frac{1}{(n+p)^2} \right).$$

Les suites  $u_n$  et  $v_n$  convergent vers  $\gamma$ , donc leur moyenne  $m_n$  converge aussi vers  $\gamma$ . À  $n$  fixé, le passage à la limite lorsque  $p \rightarrow +\infty$  donne :

$$0 \leq \gamma - m_n \leq \frac{1}{8n^2},$$

soit l'encadrement plus précis :

$$m_n \leq \gamma \leq m_n + \frac{1}{8n^2}.$$

Pour  $n = 50$ , l'erreur est majorée par  $1/20000 = 0,00005$ . On obtient exactement :

$$\sum_{j=1}^{50} \frac{1}{j} - \ln 50 - \frac{1}{100} \leq \gamma \leq \sum_{j=1}^{50} \frac{1}{j} - \ln 50 - \frac{1}{100} + \frac{1}{20000}.$$

Les bornes valent approximativement 0,577182332901 et 0,577232332901. En arrondissant vers l'extérieur, on obtient l'encadrement numérique :

$$0,57718233 \leq \gamma \leq 0,57723234.$$

Cette amélioration garantit une erreur d'ordre  $1/n^2$  au lieu de la première majoration d'ordre  $1/n$ .

## Partie I

Le tirage est uniforme sans remise. Pour  $1 \leq p \leq n$ , chaque suite de  $p$  numéros distincts a la même probabilité :

$$\frac{1}{n} \frac{1}{n-1} \cdots \frac{1}{n-p+1} = \frac{(n-p)!}{n!}.$$

Question adaptée au programme actuel : l'équiprobabilité du choix parmi les boules encore présentes est explicitée dans le sujet ; elle est nécessaire pour transformer les dénombrements en probabilités.

### 1. Probabilité d'un nouveau maximum.

(a) Choix d'une partie à  $p$  éléments.

Pour  $0 \leq p \leq n$ , le nombre de parties de  $N_n$  à  $p$  éléments est, par définition du coefficient binomial :

$$\binom{n}{p} = \frac{n!}{p!(n-p)!}.$$

Dans cette question on utilise cette formule pour  $2 \leq p \leq n$ .

(b) Ordres possibles dans une partie donnée.

Une suite de  $p$  éléments distincts appartenant à une partie  $A$  de cardinal  $p$  contient tous les éléments de  $A$ , chacun une fois. Il y a  $p$  choix pour le premier terme,  $p-1$  pour le deuxième, puis successivement jusqu'à un seul choix pour le dernier. Le nombre de suites est donc :

$$p(p-1) \cdots 1 = p!.$$

(c) Suites de  $p$  éléments distincts de  $N_n$ .

### Méthode 1 — Choix de l'ensemble, puis de l'ordre.

On choisit d'abord une partie  $A$  de cardinal  $p$ , de  $\binom{n}{p}$  façons, puis on ordonne ses éléments de  $p!$  façons d'après b). Chaque suite détermine de manière unique sa partie sous-jacente et son ordre. Il y a donc :

$$\binom{n}{p} p! = \frac{n!}{(n-p)!}$$

suites possibles.

### Méthode 2 — Choix successifs sans remise.

On choisit directement le premier terme parmi  $n$  éléments, le deuxième parmi les  $n-1$  éléments restants, et ainsi de suite. Pour le  $p$ -ième terme, il reste  $n-p+1$  choix. Le nombre de suites est :

$$n(n-1)\cdots(n-p+1) = \frac{n!}{(n-p)!}.$$

La première méthode sépare le choix des valeurs et de leur ordre ; la seconde construit immédiatement la suite.

(d) Maximum en dernière position dans une partie donnée.

Le maximum d'une partie  $A$  non vide est unique. Sa position est imposée : il est placé au rang  $p$ . Les  $p-1$  autres éléments de  $A$  peuvent occuper les positions restantes dans n'importe quel ordre. Il y a donc :

$$(p-1)!$$

suites possibles.

(e) Maximum en dernière position parmi les suites de  $N_n$ .

On choisit la partie sous-jacente  $A$ , de  $\binom{n}{p}$  façons, puis on utilise d) pour l'ordonner avec son maximum au rang  $p$ . Le nombre de suites favorables est :

$$\binom{n}{p} (p-1)! = \frac{n!}{p(n-p)!}.$$

Les parties sous-jacentes sont distinctes, donc ces choix ne comptent aucune suite deux fois.

(f) Probabilité d'un nouveau maximum au rang  $p$ .

### Méthode 1 — Dénombrement demandé par l'énoncé.

L'événement  $\{Z_p > X_{p-1}\}$  signifie que le plus grand numéro parmi les  $p$  premiers tirages occupe la dernière position. Les suites de numéros distincts sont équiprobables. Le quotient du nombre de suites favorables de e) par le nombre de suites possibles de c) vaut :

$$P(Z_p > X_{p-1}) = \frac{\binom{n}{p} (p-1)!}{\binom{n}{p} p!} = \frac{1}{p}.$$

Le dénominateur est non nul puisque  $2 \leq p \leq n$ .

### Méthode 2 — Symétrie des positions du maximum.

Pour  $1 \leq j \leq p$ , considérons l'événement selon lequel le maximum des  $p$  premiers numéros est en position  $j$ . Les numéros étant distincts, exactement une de ces positions contient le maximum : ces  $p$  événements sont disjoints et leur réunion est l'univers.

Une permutation des positions transforme bijectivement les suites équiprobables et peut échanger n'importe quelle position avec n'importe quelle autre. Les  $p$  événements ont donc la même probabilité. Leur somme valant 1, chaque probabilité vaut  $1/p$ . En particulier :

$$P(Z_p > X_{p-1}) = \frac{1}{p}.$$

Le dénombrement établit une formule explicite ; la symétrie explique pourquoi cette probabilité dépend de  $p$  et non de  $n$ .

## 2. Nombre de nouveaux maxima après le premier tirage.

La variable  $B_p$  est l'indicatrice de  $\{Z_p > X_{p-1}\}$ . Elle suit une loi de Bernoulli de paramètre  $1/p$  et, étant bornée, est intégrable. Ainsi :

$$E(B_p) = \frac{1}{p}.$$

La linéarité de l'espérance pour la somme finie donne, sans supposer l'indépendance des variables  $B_p$  :

$$E(B_2 + \dots + B_n) = \sum_{p=2}^n \frac{1}{p} = H_n - 1 = \ln n + u_n - 1.$$

La somme  $B_2 + \dots + B_n$  est le nombre de tirages, après le premier, qui établissent un nouveau maximum parmi les numéros déjà observés. Son espérance est donc le nombre moyen de mises à jour du maximum après son initialisation.

Si  $n = 1$ , la somme est vide et vaut 0 ; la formule  $H_1 - 1 = 0$  reste cohérente.

## 3. Algorithme de recherche du maximum.

Question adaptée au programme actuel. Code adapté en Python.

Pour  $n \geq 2$ ,  $Z$  contient une permutation des entiers de 1 à  $n$  ; les accès  $Z[p - 1]$  correspondent aux numéros mathématiques  $Z_p$ . Le code complet étudié est :

## Python

```
X = Z[0]
for p in range(2, n + 1):
    if Z[p - 1] > X:
        X = Z[p - 1]
```

Aucune importation n'est nécessaire. La boucle traite les positions mathématiques 2 à  $n$ . Les seules affectations comptées sont celles à  $X$ . Le code a été exécuté et contrôlé sur toutes les permutations pour les tailles de 2 à 8.

(a) Deux ordres particuliers.

Dans l'ordre croissant, l'initialisation donne  $X$  égal à 1. À chaque itération de rang  $p$ , on a  $Z_p = p > X_{p-1} = p - 1$ , donc l'affectation met  $X$  à  $p$ . Les valeurs prises lors des affectations sont :

$1, 2, \dots, n$ .

Dans l'ordre décroissant, l'initialisation donne  $X$  égal à  $n$ . Tous les numéros suivants sont inférieurs à  $n$ , donc aucune nouvelle affectation n'a lieu. La variable  $X$  garde la valeur  $n$  à toutes les étapes.

(b) Valeur finale et nombres d'opérations.

Montrons l'invariant : après traitement des  $p$  premiers éléments,  $X$  vaut

$X_p = \max(Z_1, \dots, Z_p)$ . Pour  $p = 1$ , c'est l'initialisation. Si l'invariant est vrai après  $p - 1$  éléments, le test compare  $Z_p$  à leur maximum. Si  $Z_p$  est plus grand,  $X$  reçoit  $Z_p$ ; sinon  $X$  reste inchangé. Dans les deux cas,  $X$  devient le maximum des  $p$  premiers éléments. La récurrence établit l'invariant.

Après le dernier élément,  $X$  contient donc le maximum de tous les entiers de 1 à  $n$ , soit :

$$X_n = n.$$

Il y a exactement une comparaison par itération, pour  $p = 2, \dots, n$ , donc exactement  $n - 1$  comparaisons.

L'initialisation comporte une affectation à  $X$ . Chaque itération en ajoute au plus une. Ainsi le nombre  $A_n$  d'affectations vérifie :

$$1 \leq A_n \leq n.$$

Les bornes sont atteintes : dans l'ordre décroissant on n'effectue que l'initialisation, donc le minimum est 1 ; dans l'ordre croissant chaque comparaison conduit à une mise à jour, donc le maximum est  $n$ .

Question adaptée au programme actuel : les affectations comptées sont celles de la variable de résultat  $X$ , et non les changements de l'indice de boucle. Cette précision préserve le coût étudié dans le problème.

(c) Espérance et développement asymptotique.

D'après l'invariant et la définition de  $B_p$ , une affectation supplémentaire à l'étape  $p$  a lieu exactement lorsque  $B_p = 1$ . Le nombre total d'affectations est donc :

$$A_n = 1 + \sum_{p=2}^n B_p.$$

Il est borné par  $n$ , donc intégrable. La question 2 donne :

$$E_n = E(A_n) = 1 + \sum_{p=2}^n \frac{1}{p} = H_n = \ln n + u_n.$$

Les préliminaires établissent  $u_n \rightarrow \gamma$ . En posant  $\varepsilon_n = u_n - \gamma$ , on a  $\varepsilon_n \rightarrow 0$  et :

$$E_n = \ln n + \gamma + \varepsilon_n.$$

Les coefficients demandés sont donc :

$$a = 1, \quad b = \gamma.$$

Le nombre de comparaisons est toujours  $n - 1$ , tandis que le nombre moyen d'affectations est logarithmique en  $n$ .

## Partie II

---

### 1. Probabilité d'obtenir le deuxième plus grand numéro au dernier tirage.

Fixons  $3 \leq p \leq n$ . L'événement  $\{Y_{p-1} < Z_p < X_{p-1}\}$  signifie précisément que  $Z_p$  est le deuxième plus grand élément parmi les  $p$  premiers tirages. En effet, il est inférieur au maximum des  $p - 1$  premiers et supérieur à chacun de leurs autres éléments.

Réciproquement, si  $Z_p$  est le deuxième plus grand parmi les  $p$  premiers, le plus grand appartient aux  $p - 1$  précédents et tous les autres numéros précédents sont inférieurs à  $Z_p$ .

#### Méthode 1 — Adaptation du dénombrement de I.1.

---

On choisit une partie  $A$  de cardinal  $p$ , de  $\binom{n}{p}$  façons. Son deuxième plus grand élément est déterminé et on l'impose en position  $p$ . Les  $p - 1$  autres éléments peuvent être ordonnés de  $(p - 1)!$  façons. Il y a donc  $\binom{n}{p}(p - 1)!$  suites favorables et  $\binom{n}{p}p!$  suites possibles, toutes équiprobables. Ainsi :

$$P(Y_{p-1} < Z_p < X_{p-1}) = \frac{1}{p}.$$

#### Méthode 2 — Symétrie des positions du deuxième plus grand.

---

Le deuxième plus grand élément parmi les  $p$  numéros distincts existe et est unique. Chacune des  $p$  positions est aussi probable pour cet élément, car une permutation des positions conserve l'équiprobabilité des suites. Les événements indiquant sa position sont disjoints et de réunion l'univers. Chaque position a donc une probabilité  $1/p$ , en particulier la dernière.

La méthode de dénombrement répond à la consigne de reprise de I.1 ; la symétrie donne une interprétation immédiate en termes de rang parmi les valeurs déjà tirées.

## 2. Espérance du nombre d'affectations à une étape.

Les numéros étant deux à deux distincts, les trois événements de la définition de  $C_p$  sont disjoints et forment un système complet. Il n'y a pas de cas d'égalité avec  $X_{p-1}$  ou  $Y_{p-1}$ .

La question 1 donne la probabilité du cas intermédiaire, et I.1.(f) donne celle d'un nouveau maximum. Ainsi :

$$P(C_p = 1) = \frac{1}{p}, \quad P(C_p = 2) = \frac{1}{p}, \quad P(C_p = 0) = 1 - \frac{2}{p}.$$

Comme  $p \geq 3$ , ces trois probabilités sont non négatives et leur somme vaut 1. La variable  $C_p$  est bornée par 2, donc intégrable, et :

$$E(C_p) = 0 \left(1 - \frac{2}{p}\right) + 1 \frac{1}{p} + 2 \frac{1}{p} = \frac{3}{p}.$$

Cette variable comptera les affectations aux variables X et Y lors du traitement de l'élément de rang  $p$ .

## 3. Algorithme de recherche des deux plus grands éléments.

Question adaptée au programme actuel. Code adapté en Python.

Le tableau Z contient une permutation des entiers de 1 à  $n$ , avec  $n \geq 2$ . On étudie le code complet :

### Python

```
if Z[1] > Z[0]:
    X = Z[1]
    Y = Z[0]
else:
    X = Z[0]
    Y = Z[1]
for p in range(3, n + 1):
    if Z[p - 1] > Y:
        if Z[p - 1] < X:
            Y = Z[p - 1]
```

```

else:
    Y = X
    X = Z[p - 1]

```

Aucune importation n'est nécessaire. La comparaison initiale classe les deux premiers numéros. La boucle traite ensuite les positions mathématiques 3 à  $n$  ; elle est vide pour  $n = 2$ . L'ordre des deux affectations dans la dernière branche est essentiel :  $Y$  reçoit d'abord l'ancien maximum, puis  $X$  reçoit le nouveau. Le code a été exécuté et contrôlé sur toutes les permutations pour les tailles de 2 à 8.

(a) Deux ordres particuliers.

Dans l'ordre croissant, après l'initialisation,  $X$  vaut 2 et  $Y$  vaut 1. Supposons qu'après  $p - 1$  éléments ils valent respectivement  $p - 1$  et  $p - 2$ . Le numéro suivant vaut  $p$ , donc il dépasse  $Y$  et il n'est pas inférieur à  $X$ . On exécute  $Y = X$ , puis  $X = Z[p - 1]$ . Après ces deux affectations, les valeurs sont  $p$  et  $p - 1$ .

Les couples après traitement de chaque préfixe sont donc :

$(2, 1), (3, 2), \dots, (n, n - 1)$ .

Pour préciser aussi les valeurs prises entre deux affectations, lors de l'étape  $p \geq 3$ , le couple passe de  $(p - 1, p - 2)$  à  $(p - 1, p - 1)$  après l'affectation à  $Y$ , puis à  $(p, p - 1)$  après l'affectation à  $X$ . L'égalité temporaire des deux variables ne pose pas de problème : l'invariant est attendu à la fin de l'étape.

Dans l'ordre décroissant, l'initialisation donne  $X$  égal à  $n$  et  $Y$  égal à  $n - 1$ . Tous les éléments suivants sont inférieurs à  $n - 1$  ; le premier test de chaque itération est faux, et aucune mise à jour n'a lieu. Le couple reste  $(n, n - 1)$  jusqu'à la fin.

(b) Valeurs finales et nombres extrêmes d'opérations.

Montrons l'invariant : après traitement des  $p$  premiers numéros, les variables  $X$  et  $Y$  contiennent leurs deux plus grands éléments, dans l'ordre décroissant, c'est-à-dire  $X_p$  et  $Y_p$ . Pour  $p = 2$ , cela résulte de la comparaison et des deux affectations d'initialisation.

Supposons l'invariant vrai au rang  $p - 1$ . Trois cas sont possibles puisque tous les numéros sont distincts :

- Si  $Z_p < Y_{p-1}$ , les deux plus grands éléments ne changent pas ; aucune affectation n'est exécutée.
- Si  $Y_{p-1} < Z_p < X_{p-1}$ , le maximum reste  $X_{p-1}$  et le deuxième plus grand devient  $Z_p$  ; une affectation à  $Y$  est exécutée.
- Si  $Z_p > X_{p-1}$ , le maximum devient  $Z_p$  et le deuxième plus grand devient l'ancien maximum  $X_{p-1}$  ; les deux affectations  $Y = X$  puis  $X = Z[p - 1]$  réalisent cette mise à jour.

Dans chacun des cas, l'invariant est conservé. À la fin, les deux plus grands éléments de  $\{1, \dots, n\}$  sont donc dans les variables :

La variable  $X$  contient  $n$  et la variable  $Y$  contient  $n - 1$ .

L'initialisation effectue une comparaison et deux affectations. Chacune des  $n - 2$  itérations suivantes effectue au moins une comparaison, celle avec  $Y$ , et au plus deux comparaisons, si le test avec  $X$  est également exécuté. Si  $D_n$  désigne le nombre total de comparaisons, on a :

$$n - 1 \leq D_n \leq 2n - 3.$$

L'ordre décroissant atteint le minimum  $n - 1$ , et l'ordre croissant atteint le maximum  $2n - 3$ .

Chaque itération effectue entre zéro et deux affectations aux variables du résultat. Si  $A_n$  désigne leur nombre total, on a :

$$2 \leq A_n \leq 2n - 2.$$

Le minimum 2 est atteint dans l'ordre décroissant ; le maximum  $2n - 2$  est atteint dans l'ordre croissant. Pour  $n = 2$ , la boucle est vide : minimum et maximum sont égaux à une comparaison et deux affectations, conformément aux formules.

Question adaptée au programme actuel : le nombre de comparaisons comprend les deux opérateurs  $<$  et  $>$ . Le PDF mentionne seulement  $>$  alors que le code comporte aussi un test  $<$ . Les affectations comptées sont uniquement celles à  $X$  et à  $Y$ .

(c) Espérances des coûts et développements asymptotiques.

Pour  $p \geq 3$ , posons  $J_p = 1$  si  $Z_p > Y_{p-1}$  et  $J_p = 0$  sinon. Cette indicatrice indique si la deuxième comparaison, celle avec  $X$ , est exécutée.

Les deux événements  $\{Y_{p-1} < Z_p < X_{p-1}\}$  et  $\{Z_p > X_{p-1}\}$  sont disjoints et leur réunion est  $\{Z_p > Y_{p-1}\}$ . Les questions II.1 et I.1.(f) donnent donc :

$$E(J_p) = P(Z_p > Y_{p-1}) = \frac{2}{p}.$$

Le nombre de comparaisons s'écrit exactement :

$$D_n = 1 + \sum_{p=3}^n (1 + J_p).$$

Le premier terme est la comparaison d'initialisation. À chaque étape, le terme 1 correspond au test avec  $Y$ , et  $J_p$  au test supplémentaire avec  $X$ . Toutes ces variables sont bornées, donc intégrables. La linéarité de l'espérance, sans aucune hypothèse d'indépendance supplémentaire, donne :

$$E'_n = 1 + (n-2) + 2 \sum_{p=3}^n \frac{1}{p} = n-1 + 2 \left( H_n - 1 - \frac{1}{2} \right),$$

d'où :

$$E'_n = n + 2H_n - 4 = n + 2 \ln n + 2u_n - 4.$$

Pour les affectations, l'initialisation en effectue deux. D'après les trois cas de b), chaque étape  $p$  en effectue exactement  $C_p$ . Ainsi :

$$A_n = 2 + \sum_{p=3}^n C_p.$$

La question II.2 donne alors :

$$E_n'' = 2 + 3 \sum_{p=3}^n \frac{1}{p} = 2 + 3 \left( H_n - \frac{3}{2} \right) = 3H_n - \frac{5}{2},$$

soit :

$$E_n'' = 3 \ln n + 3u_n - \frac{5}{2}.$$

Les deux expressions restent valables pour  $n = 2$  : elles donnent  $E_2' = 1$  et  $E_2'' = 2$ , car  $H_2 = 3/2$ .

Puisque  $u_n \rightarrow \gamma$ , définissons :

$$\varepsilon_n' = 2(u_n - \gamma), \quad \varepsilon_n'' = 3(u_n - \gamma).$$

Ces deux suites tendent vers 0. On obtient :

$$E_n' = n + 2 \ln n + (2\gamma - 4) + \varepsilon_n',$$

$$E_n'' = 3 \ln n + \left( 3\gamma - \frac{5}{2} \right) + \varepsilon_n''.$$

Les coefficients demandés sont donc :

$$a' = 1, \quad b' = 2, \quad c' = 2\gamma - 4, \quad a'' = 3, \quad b'' = 3\gamma - \frac{5}{2}.$$

Question adaptée au programme actuel : le reste de la seconde expression est noté  $\varepsilon_n''$ , conformément à sa limite. Le PDF écrit  $\varepsilon_n'$  dans cette expression, puis  $\varepsilon_n''$  dans la limite, ce qui est une incohérence de notation.

Le nombre moyen de comparaisons est linéaire en  $n$ , avec un terme correctif logarithmique. Le nombre moyen d'affectations aux deux variables du résultat est logarithmique en  $n$ .

## Relevé des adaptations

---

- Préliminaires 1 : explicitation de la borne  $n \geq 1$  dans la définition de  $u_n$ , imprimée de façon incomplète ; cette borne est cohérente avec la définition de  $v_n$  et le domaine des expressions.
- Préliminaires 2.(b) : demande et algorithme convertis en Python. Code adapté en Python, avec somme harmonique calculée par une boucle.
- I.1, contexte probabiliste : précision de l'équiprobabilité du choix parmi les boules restantes, nécessaire aux calculs de probabilités par dénombrement. Pour une famille finie, la notation du maximum remplace celle du supremum sans changer la définition de  $X_p$ .
- I.3 et II.3 : codes convertis en Python avec des tableaux indexés à partir de zéro et des bornes de boucle conservant exactement les positions traitées. Code adapté en Python ; précision que seules les affectations aux variables du résultat sont comptées, ainsi que les comparaisons de valeurs, à l'exclusion de la gestion des boucles et de la préparation des données.
- II.3.(b) et II.3.(c) : comptage explicite des comparaisons  $<$  et  $>$ , car le code original contient les deux opérateurs alors que la consigne de comptage ne mentionne que  $>$ .
- II.3.(c) : correction du reste de la seconde expression en  $\varepsilon_n''$ , en cohérence avec la limite annoncée et avec les deux coûts distincts.