À propos des exercices d'algorithmique du bac S 2012.

Dominique Baroux(*) & Cécile Prouteau(**)

Résumé : le groupe algorithmique de l'IREM Paris 7[1] propose quelques commentaires généraux sur les exercices d'algorithmique qui ont fait leur apparition dans les sujets du baccalauréat S de l'année 2012. Ces considérations générales sont suivies d'une analyse plus détaillée des sujets Métropole et Asie, complétée par des propositions de modifications des sujets.

Cette année l'algorithmique a fait son apparition dans les sujets du baccalauréat S : sept épreuves sur neuf au total contiennent un exercice d'algorithmique. Le sujet Antilles-Guyane en propose deux (un pour les spécialistes et un pour les non-spécialistes) et il n'y en a pas dans les sujets « Nouvelle-Calédonie » et « Liban ». Le groupe algorithmique de l'IREM Paris 7 présente dans cet article – déjà paru dans les chantiers nos 154 et 155, journal de la Régionale Ile de France – quelques commentaires généraux sur ces exercices suivis de réflexions concernant les épreuves de Métropole et d'Asie.

La plupart des algorithmes proposés dans les épreuves du Baccalauréat S de la session 2012 présentent la même structure et utilisent un vocabulaire commun. Une des conséquences pourrait être que les enseignants de lycée considèrent cette présentation d'un algorithme comme un modèle. Nous allons tenter de soulever les ambiguïtés que recèlent ces formulations.

La présentation d'un algorithme en quatre parties intitulées « Variables », « Entrée », « Traitement », « Sortie » est acceptable au niveau lycée. Mais distinguer la partie « Initialisation » de la partie « Traitement » n'a plus de réalité « technique » dans les langages modernes (ce sont juste des affectations comme les autres). Sa principale raison d'être serait d'aider les élèves à se souvenir d'initialiser les variables quand ils écrivent un algorithme.

Les formulations du type « **demander à l'utilisateur** la valeur de n », ou bien « **afficher** u », ou bien « **saisir** un réel », ou bien encore « donner la valeur exacte **affichée** par cet algorithme lorsque **l'utilisateur** entre la valeur $n = 3$ » évoquent une interface « machine » (on ne sait pas très bien d'ailleurs où cette machine se situe). Or les algorithmes proposés dans les exercices dont nous parlons peuvent tous être vus comme des procédures, ou « fonctions » au sens informatique du terme, possédant des paramètres (valeurs reçues en entrée) et un résultat dépendant de ces paramètres (valeur produite en sortie). Il n'y a donc pas de raison dans ces cas précis d'introduire dans leur écriture du vocabulaire lié à l'utilisation d'une machine

(*) domi.baroux@laposte.net

(**) cecilep.apmep.idf@free.fr

(1) Membres du groupe algorithmique Paris 7 : Dominique Baroux, Martine Bühler, Pierre Campet, Françoise Hérault, Jean-Marc Melchiore, Antoine Meyer, Cécile Prouteau.

(instructions dites « d'entrées/sorties »), qui peuvent entretenir dans l'esprit des élèves une confusion entre algorithme « idéal » et programme concret (voir les commentaires sur le sujet Métropole). Afin de renforcer cette vision d'un algorithme comme « fonction » on pourrait indiquer les paramètres et le résultat en précisant leur nature (entier, réel, …) dans les rubriques correspondantes et inverser les parties « Entrées » (ou plutôt « Paramètres ») et « Variables ».
Les algorithmes proposés sont souvent dissociés du contexte mathématique dans lequel ils sont présentés. C'est notamment le cas pour l'épreuve proposée en Asie. On peut faire l'hypothèse que des allers retours entre l'algorithme et le problème « mathématique » pourraient être bénéfiques aux élèves pour la compréhension et la résolution de l'exercice.

La suite de cet article est consacrée à l'analyse des exercices de Métropole et d'Asie. Voici le texte de l'exercice Métropole :

Partie B

Soit (u_n) la suite définie pour tout entier strictement positif par

$$u_n = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n} - \ln n.$$

1. On considère l'algorithme suivant :

Variables :	i et n sont des entiers naturels. u est un réel.
Entrée :	Demander à l'utilisateur la valeur de n.
Initialisation :	Affecter à u la valeur 0.
Traitement :	Pour i variant de 1 à n. Affecter à u la valeur $u + \frac{1}{i}$
Sortie :	Afficher u.

Donner la valeur exacte affichée par cet algorithme lorsque l'utilisateur entre la valeur $n = 3$.

2. Recopier et compléter l'algorithme précédent afin qu'il affiche la valeur de u_n lorsque l'utilisateur entre la valeur de n.

3. Voici les résultats fournis par l'algorithme modifié, arrondis à 10^{-3}

n	4	5	6	7	8	9	10	100	1000	1500	2000
u_n	0,697	0,674	0,658	0,647	0,638	0,632	0,626	0,582	0,578	0,578	0,577

À l'aide de ce tableau, formuler des conjectures sur le sens de variation de la suite (u_n) et son éventuelle convergence.

Deux collègues de l'académie de Versailles nous ont communiqué les résultats suivants qui concernent 99 copies corrigées.
Question B1 : 26 réponses correctes, 40 réponses incorrectes, 33 absences de réponse.
Question B2 : 5 réponses correctes, 50 réponses incorrectes, 44 absences de réponse.

Voici les résultats de l'Académie de Créteil à la question B2, publiés par les IPR de Créteil dans leur lettre de rentrée:

	Créteil	National
Démarche correcte	17%	18,8%
Démarche incorrecte	52%	56%
Pas de réponse	31%	25,2%

À la lecture de ces résultats, on peut légitimement s'interroger sur les raisons de ce peu de réponses correctes.

À la première question, une majorité de candidats a répondu que l'algorithme donne le terme u_3 de la suite. Cela pourrait provenir de deux facteurs : l'exercice commence par donner l'expression du terme général de la suite (u_n), puis la lettre u est utilisée dans l'algorithme. Cette présentation est susceptible d'entraîner un élève peu sûr de lui vers la faute. La formulation « Donner la valeur exacte affichée par cet algorithme lorsque l'utilisateur entre la valeur $n = 3$ » qui fait explicitement référence à une interface homme-machine a dû perturber les candidats un peu chevronnés en informatique. En effet, tel qu'il est écrit, cet algorithme implémenté sur machine ne peut donner que 1,8333333 ou toute autre valeur approchée, mais pas la valeur exacte demandée.

Les erreurs à la deuxième question auraient pu être évitées en ajoutant « Fin de Pour » à la fin de l'écriture de la boucle. En effet, les élèves ont souvent introduit une instruction supplémentaire à l'intérieur de la boucle.

Quant à la troisième question, en plus de l'apparition des valeurs approchées (voir remarque ci-dessus), il est gênant de voir apparaître u_n dans le tableau. En effet la variable u ou v de l'algorithme et les termes de la suite n'ayant pas le même statut, il est préférable de ne pas les noter de manière identique. En revanche, la donnée du tableau est intéressante car elle permet au candidat de contrôler sa réponse à la question précédente. Il peut tester l'algorithme qu'il a élaboré en l'implémentant sur sa calculatrice et éventuellement le modifier. Compte tenu du temps que cela prendrait, il faudrait attribuer au moins un point à cette question.

Pour conclure, voici une proposition de modification de cet exercice qui tient compte de toutes les remarques qui précèdent :

Partie B :

1. On considère l'algorithme suivant :

Entrée (ou paramètre) : n un entier naturel.
Variables : i entier naturel.
v nombre réel.
Traitement : Affecter à v la valeur 0.
Pour i variant de 1 à n
Affecter à v la valeur $v+\dfrac{1}{i}$
Fin de Pour
Sortie (ou résultat) : v.

a. Quel nombre obtient-on en sortie de l'algorithme pour $n = 3$? On en donnera la valeur exacte.
b. Quel nombre obtient-on en sortie de l'algorithme pour n quelconque supérieur ou égal à 1 ?

2. Soit la suite définie pour tout entier strictement positif par :

$$u_n = 1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{n} - \ln n.$$

Recopier et compléter l'algorithme précédent afin qu'il retourne le terme u_n lorsqu'on entre l'entier n.

3. La mise en œuvre de l'algorithme modifié sur une « machine » a donné ces résultats, arrondis à 10^{-3}.

n	4	5	6	7	8	9	10	100	1000	1500	2000
Résultat	0,697	0,674	0,658	0,647	0,638	0,632	0,626	0,582	0,578	0,578	0,577

À l'aide de ce tableau, formuler des conjectures sur le sens de variation de la suite (u_n) et son éventuelle convergence.

Voici le texte de l'exercice 4 du sujet de Bac de Terminale S Asie juin 2012

1. On considère l'algorithme suivant :

Entrée	Saisir un réel strictement positif non nul a Saisir un réel strictemenl positif non nul b $(b > a)$ Saisir un entier naturel non nul N
Initialisation	Affecter à u la valeur a Affecter à v la valeur b Affecter à n la valeur 0
Traitement	TANTQUE $n < N$ Affecter à n la valeur $n+1$ Affecter à u la valeur $\frac{a+b}{2}$ Affecter à v la valeur $\sqrt{\frac{a^2+b^2}{2}}$ Affecter à a la valeur u Affecter à b la valeur v
Sortie	Afficher u, afficher v

Reproduire et compléter le tableau suivant, en faisant fonctionner cet algorithme pour $a = 4, b = 9$ et $N = 2$. Les valeurs successives de u et v seront arrondies au millième.

n	a	b	u	v
0	4	9		
1				
2				

Dans la suite, a et b sont deux réels tels que $0 < a < b$.

On considère les suites (u_n) et (v_n) définies par :

$u_0 = a$, $v_0 = b$ et, pour tout entier naturel n :

$$u_{n+1} = \frac{u_n + v_n}{2} \quad \text{et} \quad v_{n+1} = \sqrt{\frac{u_n^2 + v_n^2}{2}}$$

2. a. Démontrer par récurrence que, pour tout entier naturel n, on a : $u_n > 0$ et $v_n > 0$.
   b. Démontrer que, pour tout entier naturel n : $v_{n+1}^2 - u_{n+1}^2 = \left(\frac{u_n - v_n}{2}\right)^2$.
   En déduire que, pour tout entier naturel n, on a $u_n \leqslant v_n$.
3. a. Démontrer que la suite (u_n) est croissante.
   b. Comparer v_{n+1}^2 et v_n^2. En déduire le sens de variation de la suite (v_n).
4. Démontrer que les suites (u_n) et (v_n) sont convergentes.

Concernant le fond, ce sujet présente quelques points positifs. Premièrement, contrairement à l'exercice 3 du sujet de métropole de juin 2012, il commence par l'algorithme. Il présente également un contenu mathématique conséquent avec l'étude de deux suites jumelles adjacentes ce qui fait apparaître du point de vue algorithmique la nécessité d'utiliser une troisième variable. L'indépendance des différentes questions et en particulier des parties algorithmiques et mathématiques est positive car elle permet de ne pas risquer de pénaliser les élèves. Cependant, il aurait été pertinent de poser à la fin de l'exercice une question permettant de montrer si l'élève a compris le lien entre les parties mathématiques et algorithmiques Il aurait aussi été intéressant d'ajouter une question sur les conditions d'arrêt de la boucle « TANTQUE ».
En ce qui concerne la forme, comme pour le sujet Métropole de juin 2012, les mots « saisir » et « afficher », bien que couramment utilisés dans les manuels, nous semblent malvenus. Il nous paraît effectivement important de dissocier la notion d'algorithme de celle de programmation et d'interface homme-machine.

Dans la partie initialisation de l'algorithme, affecter des valeurs à u et v n'a aucune utilité. En effet, avec n initialisé à 0 et N entier strictement positif, la boucle sera lancée au moins une fois et des valeurs seront alors affectées à u et v. Le tableau à compléter est quant à lui très ambigu ! Lors de la boucle « TANTQUE », les valeurs de n, a, b, u et v sont modifiées. Quelles sont les valeurs demandées ? Celles de début de boucle ? Celles de fin de boucle ? Il aurait été souhaitable de le préciser dans le tableau pour éviter de perturber les élèves. Une colonne sur le test à remplir à chaque étape aurait également pu être ajoutée afin d'aider les élèves à la mise en œuvre de l'algorithme. L'ordre dans lequel les variables apparaissent dans le tableau serait également à modifier pour plus de clarté. Le choix de u et v comme variables alors que les termes des suites se nomment u_n et v_n peut être également critiqué car il apporte une confusion quant à ce nouveau statut de lettre abordé en algorithmique. Voici une proposition de modification de cet énoncé qui tient compte des remarques précédentes.

1. On considère l'algorithme suivant :

Entrée	Un réel strictement positif non nul a Un réel strictemenl positif non nul b ($b > a$) Un entier naturel non nul N
Traitement	Affecter à n la valeur 0 TANTQUE $n < N$ Affecter à n la valeur $n+1$ Affecter à r la valeur $\dfrac{a+b}{2}$ Affecter à s la valeur $\sqrt{\dfrac{a^2+b^2}{2}}$ Affecter à a la valeur r Affecter à b la valeur s Fin de TANTQUE
Sortie	la valeur r, la valeur s

Reproduire et compléter le tableau suivant, en faisant fonctionner cet algorithme pour $a = 4, b = 9$ et $N = 2$. Les valeurs successives de r et s seront arrondies au millième.

Etape	N	n	r	s	a	b	Test $n<N$?
Entrée	2	-	-	-	4	9	-
Début itération n°1	2	0					0<2 ? vrai
Fin itération n°1							
Fin itération n°2							
...							

2. A-t-on la certitude de sortir de la boucle "TANTQUE" ? Justifiez votre réponse.

Dans la suite, a et b sont deux réels tels que $0 < a < b$.
On considère les suites (u_n) et (v_n) définies par :
$u_0 = a, v_0 = b$ et, pour tout entier naturel n :

$$u_{n+1} = \frac{u_n + v_n}{2} \quad \text{et} \quad v_{n+1} = \sqrt{\frac{u_n^2 + v_n^2}{2}}$$

3. a. Démontrer par récurrence que, pour tout entier naturel n, on a : $u_n > 0$ et $v_n > 0$.

   b. Démontrer que, pour tout entier naturel n : $v_{n+1}^2 - u_{n+1}^2 = \left(\dfrac{u_n - v_n}{2}\right)^2$.
   En déduire que, pour tout entier naturel n, on a $u_n \leqslant v_n$.

4. a. Démontrer que la suite (u_n) est croissante.

   b. Comparer v_{n+1}^2 et v_n^2. En déduire le sens de variation de la suite (v_n).

5. Démontrer que les suites (u_n) et (v_n) sont convergentes.

6. Quel lien peut-on établir entre l'algorithme de la première question et les suites (u_n) et (v_n) ?

L'usage de *a* et *b* dans la boucle peut être discuté. En effet, dans le sujet original, ces valeurs entrées comme des constantes finissent par servir de paramètres. Vu que *a* et *b* ont un sens particulier dans le reste de l'exercice, une solution serait de déclarer deux variables *res1* et *res2* pour le résultat du calcul, et deux variables temporaires *t1* et *t2* pour le calcul intermédiaire. Cette solution serait moins efficace d'un point de vue gestion mémoire, mais l'efficacité ne nous semble pas être le problème majeur ici. On aurait alors l'avantage d'éviter toute ambiguïté sur le nom des variables, et de ne pas modifier les valeurs de *a* et *b* , qui garderaient leur rôle de constantes (comme dans le reste du texte).

Le choix du type de la boucle est également discutable puisque l'on connaît le nombre d'itérations. On apprend aux élèves que la boucle « TANTQUE » est intéressante lorsque l'on ne connaît pas à l'avance le nombre d'itérations. Ici, une boucle « POUR *i* allant de 1 à N » aurait été beaucoup plus simple et aurait évité par exemple le problème du test dans le tableau. Nous avons fait le choix de garder la boucle « TANTQUE » proposée dans le sujet original, car même si l'algorithme proposé dans ce sujet n'est pas idéal, il permet d'évaluer les élèves sur le fonctionnement d'une telle boucle et l'utilisation des tests. Une question aurait pu être ajoutée à la fin du sujet concernant le choix de cette boucle. Par exemple : « Quel autre type de boucle auriez-vous pu utiliser en remplacement de la boucle « TANTQUE » ? ». Il est également possible d'étendre l'exercice en classe en proposant une critique de l'algorithme et une réécriture avec une boucle « POUR *i* allant de 1 à N ».
Pour aller plus loin, il est aussi envisageable de proposer, suite à cet exercice, une autre étude de suites jumelles en demandant à la fin aux élèves d'écrire eux-mêmes l'algorithme puis de leur faire remarquer qu'il s'agit du même. Une généralisation pourrait ensuite être apportée en classe en écrivant un algorithme ayant en entrée les deux fonctions permettant de définir les suites jumelles u_n et v_n.

Des compléments sur le sujet Asie sont présentés sur le site internet de la Régionale Ile de France à l'adresse www.apmep-iledefrance.org. D'autres versions modifiées de sujet sont téléchargeables sur ce même site.

L'algorithmique est un nouvel objet à intégrer dans les programmes de mathématiques, et cette intégration ne va pas de soi pour les enseignants de mathématiques. Les mots et notations utilisés dans ces sujets de baccalauréat véhiculent des confusions de natures diverses qui pourraient être préjudiciables à l'enseignement des notions d'algorithmique. Afin d'aider les enseignants de mathématiques dans la mise en œuvre de cet enseignement il nous a paru essentiel de soulever et d'expliciter les ambiguïtés qui se cachent derrière certaines formulations. D'autre part, différents témoignages montrent que les enseignants de mathématiques s'interrogent sur la légitimité d'un apprentissage en algorithmique dans l'enseignement secondaire. Dans ce contexte, il est important de montrer les atouts du point de vue algorithmique dans l'acquisition des notions mathématiques et dans la résolution des problèmes, et cela chaque fois que c'est possible. C'est dans cet esprit que nous avons systématiquement proposé des liens entre la situation mathématique et l'algorithme.