Découpage en lignes d'un paragraphe de texte

Épreuve pratique d'algorithmique et de programmation
Concours commun des Écoles normales supérieures

Durée de l'épreuve: 3 heures 30 minutes

Juin/Juillet 2016

ATTENTION !

N'oubliez en aucun cas de recopier votre u_0
à l'emplacement prévu sur votre fiche réponse

Important.

Il vous a été donné un numéro u_0 qui servira d'entrée à vos programmes. Les réponses attendues sont généralement courtes et doivent être données sur la fiche réponse fournie à la fin du sujet. À la fin du sujet, vous trouverez en fait deux fiches réponses. La première est un exemple des réponses attendues pour un $\widetilde{u_0}$ particulier (précisé sur cette même fiche et que nous notons avec un tilde pour éviter toute confusion!). Cette fiche est destinée à vous aider à vérifier le résultat de vos programmes en les testant avec $\widetilde{u_0}$ au lieu de u_0. Vous indiquerez vos réponses (correspondant à votre u_0) sur la seconde et vous la remettrez à l'examinateur à la fin de l'épreuve.

En ce qui concerne la partie orale de l'examen, lorsque la description d'un algorithme est demandée, vous devez présenter son fonctionnement de façon schématique, courte et précise. Vous ne devez en aucun cas recopier le code de vos procédures!
Quand on demande la complexité en temps ou en mémoire d'un algorithme en fonction d'un paramètre n, on demande l'ordre de grandeur en fonction du paramètre, par exemple: $O(n^2)$, $O(n \log n)$,...
Il est recommandé de commencer par lancer vos programmes sur de petites valeurs des paramètres et de ***tester vos programmes sur des petits exemples que vous aurez résolus préalablement à la main ou bien à l'aide de la fiche réponse type fournie en annexe***.
Enfin, il est recommandé de lire l'intégralité du sujet avant de commencer afin d'effectuer les bons choix de structures de données dès le début.

1 Production de texte

Si $a \geqslant 0$ et $b \geqslant 1$ sont deux entiers, on note $a \bmod b$ le reste de la division euclidienne de a par b, autrement dit l'unique entier r avec $0 \leqslant r < b$ tel qu'il existe un entier q satisfaisant $a = bq + r$.

À partir de la valeur initiale u_0 qui vous a été donnée, on définit une suite d'entiers $(u_n)_{n\geqslant 0}$ par

$$u_{n+1} = (1\,000\,001\,u_n + 3) \bmod 2^{30} \qquad (\text{on a } 2^{30} = 1\,073\,741\,824).$$

Question 1 *Calculer $u_n \bmod 10\,000$ pour* ***a)*** $n = 5$ ***b)*** $n = 1\,000$ ***c)*** $n = 10\,000\,000$.

Soit $f : \mathbb{N} \to \mathbb{N}$ la fonction définie par

$$f(\ell) = \begin{cases} \lfloor (\ell+4)\, 3^\ell\, 2^{30-2(\ell+1)} \rfloor & \text{si } 0 \leqslant \ell < 30, \\ 0 & \text{si } \ell \geqslant 30, \end{cases} \tag{1}$$

où la notation $\lfloor \cdot \rfloor$ désigne la partie entière.

Question à développer pendant l'oral 1 *Expliquer comment calculer $f(\ell)$ pour $\ell \in \mathbb{N}$ à l'aide d'opérations arithmétiques sur des entiers de 63 bits sans provoquer de dépassement de capacité. On rappelle qu'un entier signé de 63 bits permet de représenter les valeurs comprises (au sens large) entre -2^{62} et $2^{62} - 1$.*

On admet qu'un calcul direct de $f(\ell)$ en arithmétique en virgule flottante double précision à l'aide de la formule (1) fournit aussi le résultat exact, malgré les erreurs d'arrondi possibles.

Question 2 *Calculer*
a) $f(u_{10} \bmod 30) \bmod 9\,973$, ***b)*** $f(u_{20} \bmod 30) \bmod 9\,973$, ***c)*** $\sum_{i=0}^{29} (u_i \cdot f(i)) \bmod 9\,973$.

Notons $\mathcal{A}$ l'alphabet formé des caractères suivants : les 26 lettres $\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}$, les signes de ponctuation virgule **,** et point **.**, et le blanc noté $\sqcup$. Un texte est une suite finie d'éléments [1] de $\mathcal{A}$. Un mot d'un texte t est un facteur non vide de t formé de lettres et qui n'est ni précédé ni suivi par une lettre [2].

On définit par récurrence une suite de caractères $(a_n) \in \mathcal{A}^{\mathbb{N}}$ de la façon suivante.

- Si $n = 0$, si a_{n-1} est un blanc, ou si a_{n-1} est une lettre et la longueur ℓ du dernier mot de $a_0 \ldots a_{n-1}$ vérifie $u_n < f(\ell)$, alors a_n est une lettre déterminée par la valeur de $p = u_n \bmod 26$: on pose $a_n = \mathtt{a}$ si $p = 0$, $a_n = \mathtt{b}$ si $p = 1$, ..., $a_n = \mathtt{z}$ si $p = 25$;
- si a_{n-1} est un signe de ponctuation, on pose $a_n = \sqcup$;
- dans les autres cas, on pose

$$a_n = \begin{cases} . & \text{si } u_n \bmod 17 = 0 \\ , & \text{si } u_n \bmod 17 = 1 \\ \sqcup & \text{sinon.} \end{cases}$$

1. C'est-à-dire un « mot sur $\mathcal{A}$ » dans la terminologie usuelle des langages formels, mais nous utilisons ici le terme « mot » dans le sens de « suite de lettres séparées par des espaces ou des signes de ponctuation ».

2. Plus formellement, posons $\mathcal{L} = \{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\} \subseteq \mathcal{A}$. Si $t = a_0 a_1 \ldots a_{n-1} \in \mathcal{A}^*$, un mot de t de longueur ℓ est un texte de la forme $a_m a_{m+1} \ldots a_{m+\ell-1}$ tel que l'on ait (i) $\ell \geqslant 1$, (ii) $a_{m+i} \in \mathcal{L}$ pour $0 \leqslant i < \ell$, (iii) $m = 0$ ou $a_{m-1} \notin \mathcal{L}$, et (iv) $m + \ell = n$ ou $a_{m+\ell} \notin \mathcal{L}$.

Pour $n \geqslant 1$, on note t_n le texte "$a_0 a_1 \ldots a_{n-2}$." (le dernier caractère est toujours un point). Par exemple, le texte t_{80} est

```
qjgne,␣bsbgl␣rs␣ejo␣cjgfm␣yr␣bqf␣lwp␣lmj.␣otgfw␣cvk.␣lkf␣hgv␣vezyh␣vobgbct␣rgfg.
```

lorsque $u_0 = 42$.

Question 3 *Pour chacun des couples (c, n) suivants, calculer le nombre d'occurrences du caractère c dans le texte t_n et la position dans le texte (comptée à partir de zéro) de la dernière, si elle existe. On donnera tous les résultats modulo 9 973.* ***a)*** $n = 500$, $c =$ `a` ; ***b)*** $n = 1\,000$, $c =$ `.` ; ***c)*** $n = 1\,000\,000$, $c =$ `␣` ; ***d)*** $n = 1\,000\,000$, $c =$ `,` ; ***e)*** $n = 1\,000\,000$, $c =$ `z`.

Question à développer pendant l'oral 2 *Montrer que les textes t_n ne comportent aucun mot de plus de 30 lettres.*

2 Découpage en lignes avec des caractères de largeur fixe

On s'intéresse dans cette partie au découpage d'un texte t en lignes tenant sur un nombre donné c de colonnes, avec des caractères de largeur fixe. Les règles de coupure sont les suivantes.

1. On peut passer à la ligne au niveau d'un blanc. Dans ce cas, le blanc en question « disparaît » du texte : il ne contribue ni à la longueur de la ligne qu'il termine, ni à celle de la ligne suivante. (Si plusieurs blancs se suivent, seul celui où l'on passe à la ligne disparaît.)
2. On peut passer à la ligne entre une voyelle (`a`, `e`, `i`, `o`, `u`, ou `y`) et une consonne qui la suit immédiatement, en insérant entre les deux un tiret de césure (noté `-`), à condition que le mot coupé comporte au moins deux lettres avant le tiret et deux lettres après. Le tiret se place avant le retour à la ligne ; il occupe un caractère et contribue à la longueur de la ligne qu'il termine.
3. À l'issue du découpage, chaque ligne doit comporter au moins un et au plus c caractères. La concaténation des lignes après restauration des blancs supprimés en application de la règle 1 et suppression des tirets insérés en application de la règle 2 doit redonner le texte t.

Voici par exemple un découpage du texte t_{80} correspondant à $u_0 = 42$ en trois lignes de $c = 36$ colonnes :

```
012345678901234567890123456789012345
qjgne,␣bsbgl␣rs␣ejo␣cjgfm␣yr␣bqf␣lwp
lmj.␣otgfw␣cvk.␣lkf␣hgv␣vezyh␣vo-
bgbct␣rgfg.
012345678901234567890123456789012345
```

Question à développer pendant l'oral 3 *Estimer grossièrement le nombre de façons de découper un texte de longueur n en lignes de $c = 80$ colonnes quand n tend vers l'infini. (La croissance de ce nombre avec n est-elle linéaire, quadratique, polynomiale... ?)*

Question 4 *Donner le nombre de positions dans le texte t_n où il est possible de couper un mot en application de la règle 2 ci-dessus, pour* ***a)*** $n = 70$, ***b)*** $n = 10\,000$, ***c)*** $n = 1\,000\,000$.

On cherche maintenant à déterminer la « dernière » position de passage à la ligne autorisée « avant » le premier caractère qui « dépasse » de la ligne, et la nature (passage à la ligne entre deux mots ou césure d'un mot) de la coupure de ligne correspondante. Plus précisément,

nous noterons $P(t, c)$ le nombre maximal de caractères de t que peut contenir la première ligne d'un découpage de t en lignes de c colonnes. Par exemple, pour $u_0 = 42$, on a $P(t_{80}, 36) = 36$, $P(t_{80}, 35) = 32$, et $P(t_{80}, 72) = 69$.

Question 5 *On pose dans cette question* $p_i = P(t_{2000}, 80 + (u_i \bmod 1000))$. *Calculer, s'ils existent (écrire* $\varnothing$ *sinon)* ***a)*** *le plus petit entier* i *avec* $0 \leqslant i < 200$ *tel que* p_i *soit une position dans* t *où l'on peut passer à la ligne entre deux mots (règle 1 ci-dessus), et la valeur* p_i *correspondante ;* ***b)*** *le plus petit entier* i *avec* $0 \leqslant i < 200$ *tel que* p_i *soit une position dans* t *où l'on peut passer à la ligne en coupant un mot (règle 2 ci-dessus), et la valeur* p_i *correspondante ;* ***c)*** *la somme modulo* 9 973 *des* p_i *pour* $0 \leqslant i < 200$.

On note $S_c(t)$ le nombre minimal de lignes nécessaires pour écrire le texte t sur c colonnes.

Question à développer pendant l'oral 4 *Décrire un algorithme qui calcule un découpage d'un texte* t *en* $S_c(t)$ *lignes de* c *colonnes. Démontrer que votre algorithme est correct. Analyser sa complexité en fonction de* n.

Question 6 *Calculer* ***a)*** $S_{40}(t_{200})$, ***b)*** $S_{80}(t_{10\,000})$, ***c)*** $S_{40}(t_{1\,000\,000})$.

Question 7 *Donner la valeur modulo* 9 973 *du nombre de façons différentes de découper le texte* t_n *en lignes tenant sur* c *colonnes* ***a)*** *pour* $n = 41$ *et* $c = 40$, ***b)*** *pour* $n = 10\,000$ *et* $c = 40$, ***c)*** *pour* $n = 100\,000$ *et* $c = 1\,000$.

Question à développer pendant l'oral 5 *Expliquer l'algorithme que vous avez utilisé pour répondre à la question précédente. Analyser sa complexité en fonction de* n *et* c.

3 Justification à la Knuth & Plass

Nous étudions dans cette partie une variante simplifiée de l'algorithme de coupure de ligne du système typographique TEX. Il s'agit de justifier un paragraphe de texte en répartissant l'espace excédentaire entre les mots. On cherche à éviter à la fois les lignes serrées, où les mots sont presque accolés, et les lignes lâches, avec trop de blanc.

Les règles de coupure du texte sont les mêmes que dans la partie précédente (avec $c = \infty$), mais les caractères sont maintenant de taille variable.

— Les blancs sont des ressorts ; ils ont une largeur naturelle $w(\sqcup)$, mais peuvent s'étirer ou rétrécir. L'étirement et la compression acceptables sont contrôlés par deux paramètres $y(\sqcup)$ et $z(\sqcup)$. *Sauf mention contraire*, nous prendrons $w(\sqcup) = 6$, $y(\sqcup) = 3$ et $z(\sqcup) = 2$. On adjoint de plus à l'alphabet $\mathcal{A}$ un ressort spécial, noté $\leftrightsquigarrow$, qui vérifie $w(\leftrightsquigarrow) = 0$, $y(\leftrightsquigarrow) = 1\,000\,000\,000$ et $z(\leftrightsquigarrow) = 0$. Du point de vue des règles de coupure de ligne, $\leftrightsquigarrow$ se comporte comme un signe de ponctuation. On note $\mathcal{A}_{\leftrightsquigarrow} = \mathcal{A} \cup \{\leftrightsquigarrow\}$, et on appelle encore texte un élément de $\mathcal{A}_{\leftrightsquigarrow}^*$.

— Les lettres et signes de ponctuation sont modélisés par des boîtes dont la largeur $w(a)$ dépend de la taille du caractère dans la fonte utilisée. Nous supposerons ici $w(a) = 9$ quand a est une lettre et $w(a) = 5$ quand a est un signe de ponctuation. Le tiret de césure est lui aussi une boîte de largeur $w(\text{-}) = 6$. On pose $y(b) = z(b) = 0$ pour toute boîte b.

Supposons fixée une longueur de ligne L. On modélise une ligne par un couple $x = (x_0 \dots x_{n-1}, b_x)$ où $x_0 \dots x_{n-1} \in \mathcal{A}^*_{\leftrightsquigarrow}$ est un texte et $b_x \in \{\text{vrai}, \text{faux}\}$ indique si la ligne doit se terminer par un tiret de césure. Étant donné un tel couple, on définit la largeur naturelle $W(x)$, l'extensibilité totale $Y(x)$ et la compressibilité totale $Z(x)$ de x par

$$W(x) = \left(\sum_{i=0}^{n-1} w(x_i)\right) + w', \qquad Y(x) = \sum_{i=0}^{n-1} y(x_i), \qquad Z(x) = \sum_{i=0}^{n-1} z(x_i), \tag{2}$$

où $w' = w(\text{-})$ si $b_x = \text{vrai}$ et $w' = 0$ sinon. On appelle ajustement de x la quantité

$$r(x) = \begin{cases} 0 & \text{si } W(x) = L, \\ (L - W(x))/Y(x) & \text{si } W(x) < L \text{ et } Y(x) > 0, \\ (L - W(x))/Z(x) & \text{si } W(x) > L \text{ et } Z(x) > 0, \\ -\infty & \text{sinon.} \end{cases} \tag{3}$$

Enfin, on pose

$$\lambda(\rho, \beta) = \big(1 + 100\,\rho^3 + \beta\big)^2, \tag{4}$$

et l'on appelle laideur de x la quantité

$$\begin{cases} \lambda\big(|r(x)|, 50\big) & \text{si } -1 < r(x) < 10 \text{ et } b_x = \text{vrai}, \\ \lambda\big(|r(x)|, 0\big) & \text{si } -1 < r(x) < 10 \text{ et } b_x = \text{faux}, \\ \infty & \text{sinon.} \end{cases} \tag{5}$$

Pour calculer les laideurs, on appliquera les formules (2) à (5) de manière approchée, en tronquant le résultat de chaque opération élémentaire ($+$, $-$, $\times$, $/$) à trois chiffres décimaux après la virgule [3]. Par exemple, on retient comme résultat de la multiplication $0{,}049 \times (-0{,}222) = -0{,}010878$ la valeur approchée $-0{,}010$.

Question à développer pendant l'oral 6 *Montrer que la multiplication approchée ainsi définie n'est pas associative. Expliquer comment représenter les valeurs approchées en utilisant des entiers et comment réaliser chacune des opérations suivantes : addition, multiplication, division par un entier.*

On notera que dans la formule (4), l'opération d'élévation au cube est prioritaire sur la multiplication et doit donc être effectuée avant.

Question 8 *Calculer en utilisant l'arithmétique approchée définie ci-dessus :*

a) $\displaystyle\sum_{i=0}^{9} \frac{(-1)^i}{1 + (u_i \bmod 30)}$, ***b)*** $\displaystyle\sum_{i=0}^{999} \lambda\Big(\frac{u_i \bmod 10\,000}{1\,000}, u_{i+1} \bmod 50\Big)$.

Question 9 *Calculer la laideur de la ligne (t_{60}, b) pour chacun des jeux de paramètres suivants.* ***a)*** *$L = 300$, $y(\text{␣}) = 3$ et $b = \text{faux}$,* ***b)*** *$L = 500$, $y(\text{␣}) = 3$ et $b = \text{faux}$,* ***c)*** *$L = 500$, $y(\text{␣}) = 6$ et $b = \text{vrai}$.*

La laideur d'un découpage en lignes est la somme des laideurs des lignes. On considérera qu'un découpage en lignes est d'autant meilleur esthétiquement que sa laideur est petite. La laideur minimale d'un texte $t \in \mathcal{A}^*_{\leftrightsquigarrow}$ est la laideur minimale d'un découpage en lignes de t. Pour $t \in \mathcal{A}^*$, on note $t^{\leftrightsquigarrow}$ le texte de $\mathcal{A}^*_{\leftrightsquigarrow}$ formé en ajoutant un symbole $\leftrightsquigarrow$ à la fin de t.

3. Autrement dit en l'arrondissant au multiple de 10^{-3} *de valeur absolue inférieure ou égale* le plus proche.

Question à développer pendant l'oral 7 *Quelle est la laideur minimale pour un découpage en s lignes, et dans quelle situation est-elle atteinte ? Quel est l'intérêt de remplacer un texte $t \in \mathcal{A}^*$ par $t^{\leftrightsquigarrow}$ quand on en recherche un découpage de laideur minimale ?*

Question à développer pendant l'oral 8 *Exprimer la laideur minimale d'un texte $t \in \mathcal{A}^*_{\leftrightsquigarrow}$ en fonction des laideurs minimales de ses préfixes. On s'efforcera de limiter le nombre de préfixes dont il est nécessaire de connaître la laideur pour déterminer celle de t.*

Question 10 *Pour chacun des couples (n, L) suivants, donner la laideur minimale de $t_n^{\leftrightsquigarrow}$ pour un découpage en lignes de longueur L :* ***a)*** *$n = 400$, $L = 500$;* ***b)*** *$n = 1\,000$, $L = 500$;* ***c)*** *$n = 10\,000\,000$, $L = 2\,000$;* ***d)*** *$n = 1\,000\,000$, $L = 20\,000$.*

Question à développer pendant l'oral 9 *Expliquer l'algorithme que vous avez utilisé pour répondre à la question précédente, ainsi que les éventuelles idées d'améliorations que vous auriez. Analyser sa complexité en fonction de n et L. Cet algorithme se généralise-t-il à la mise en forme de texte avec des lignes de longueur variable (devant s'adapter à la forme d'une figure, par exemple) ?*

4 Découpage dynamique

On revient maintenant dans cette partie au modèle à caractères de largeur fixe de la partie 2, et l'on cherche à mettre à jour le découpage en lignes au fur et à mesure de l'édition d'un texte. Pour $0 \leqslant p < n - 1$ et $q \geqslant 0$, soit

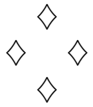
$$t'_n(p, q) = \text{“}a_0 a_1 \dots a_{p-1} a_0 a_1 \dots a_{q-1} a_p a_{p+1} \dots a_{n-2}\,.\text{”}$$

le texte obtenu en insérant le texte t_{q+1} privé de son point final dans t_n à la position p.

Question 11 *Donner le nombre de valeurs de k telles que*

$$S_{80}(t'_n(u_k \bmod (n-1), u_{k+1} \bmod 100)) > S_{80}(t_n)$$

pour ***a)*** *$n = 1\,000$ et $0 \leqslant k < 20$,* ***b)*** *$n = 10$ et $0 \leqslant k < 1\,000$,* ***c)*** *$n = 10\,000$ et $0 \leqslant k < 10\,000$,* ***d)*** *$n = 100\,000$ et $0 \leqslant k < 100\,000$,* ***e)*** *$n = 10\,000\,000$ et $0 \leqslant k < 200\,000$.*



Fiche réponse type: Découpage en lignes d'un paragraphe de texte

$\widetilde{u_0}$: 42

Question 1

a) 9833

b) 6098

c) 6618

Question 2

a) 4074

b) 6506

c) 6275

Question 3

a) $16,494$

b) $14,999$

c) $5652,2695$

d) $1488,2641$

e) $9965,2693$

Question 4

a) 1

b) 551

c) 57522

Question 5

a) $0,120$

b) $4,909$

c) 4960

Question 6

a) 6

b) 128

c) 25688

Question 7

a) 255

b) 3304

c) 4215

Question 8

a) 0.671

b) 1357762217259.800

Question 9

a) $+\infty$

b) 21.160

c) 2601.000

Question 10

a) 346.680

b) 8521.090

c) 56344.300

d) 415.000

Question 11

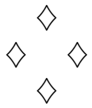
a) 19

b) 292

c) 2824

d) 57972

e) 119923



Fiche réponse: Découpage en lignes d'un paragraphe de texte

Nom, prénom, u_0: .....

Question 1

a)

b)

c)

Question 2

a)

b)

c)

Question 3

a)

b)

c)

d)

e)

Question 4

a)

b)

c)

Question 5

a)

b)

c)

Question 6

a)

b)

c)

Question 7

a)

b)

c)

Question 8

a)

b)

Question 9

a)

b)

c)

Question 10

a)

b)

c)

d)

Question 11

a)

b)

c)

d)

e)