Décimales de π

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

Durée de l'épreuve : 3 heures 30 minutes

Juin/Juillet 2014

ATTENTION !

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

Important.

Sur votre table est indiqué 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 Introduction

Nous nous intéressons au problème de calculer des décimales du nombre π en utilisant très peu de mémoire et de temps, en base 16 (questions 1 à 4) puis en base 10 (questions 5 à 8). Les deux parties du sujet sont presque indépendantes, même si les idées des algorithmes du calcul des décimales dans ces deux bases (questions 4 et 9) sont proches. En particulier, les questions 5, 6 et 7 peuvent être réalisées à partir des questions 1 et 2.

On notera $\{x\} = x - \lfloor x \rfloor$ la partie fractionnaire d'un réel x avec $\lfloor x \rfloor$ la partie entière inférieure.
Considérons $(u_k)_{k\geq 0}$ la suite d'entiers définie par :

$$u_k = \begin{cases} \text{votre } u_0 \quad (\textit{À reporter sur votre fiche réponse}) & \text{si } k = 0 \\ 4909 \times u_{k-1} \mod 120791 & k > 0. \end{cases}$$

Pour résoudre ce problème, nous aurons besoin de convertir des nombres en base 16 et de trouver la décomposition binaire d'un entier.

1.1 Représentation en base 16 et en binaire

Programmer un algorithme calculant la représentation en base 16 d'un nombre en virgule flottante et un autre donnant la représentation binaire d'un *entier*. Ce dernier écrira les bits des bits de poids faibles vers les bits de poids forts : $b = b_0 b_1 \dots b_n$ pour $b = \sum_{i=0}^{n} b_i 2^i$. Donner l'écriture en base 16 des flottants a) et b) et l'écriture en binaire des entiers c) et d).

Question 1 ***a)*** $u_{10}/1000$ ***b)*** $u_{100}/10000$ ***c)*** $u_{10} \bmod 1000$ ***d)*** $u_{100} \bmod 1000$.

Question à développer pendant l'oral : Décrire vos algorithmes ainsi que leur complexité.

1.2 Multiplication et Exponentiation modulaires en base 2

La multiplication modulaire $a \times b \bmod m$ et l'exponentiation modulaire $a^b \bmod m$ binaire utilisent la décomposition de b en base 2 pour calculer rapidement ces entiers.

Question à développer pendant l'oral : Écrire une formule pour calculer la multiplication et une autre pour calculer l'exponentiation modulaire en fonction des bits b_i de $b = \sum_{i=0}^{n} b_i 2^i$, de a et m. Expliquer comment calculer rapidement $a \times 2^i \bmod m$ et $a^{2^i} \bmod m$. Décrire vos algorithmes et donner leur complexité. On justifiera pourquoi les calculs sont exacts si $m < 2^{30}$ et que les entiers sont codés sur 31 bits.

Calculer le résultat des multiplications modulaires des deux premières entrées modulo la troisième dans les questions a) et b) et l'exponentiation modulaire de la première à la puissance la seconde modulo la troisième dans les questions c) et d).

Question 2 ***a)*** $u_{100}, u_{200}, 40289$ ***b)*** $u_{1000}, u_{1500}, 120791$ ***c)*** $u_{100}, u_{200}, 40289$ ***d)*** $u_{1000}, u_{1500}, 120791$.

2 Décimales de π en base 16

La formule suivante dûe à Bailey, Borwein et Plouffe (BBP) permet de calculer les décimales de π en base 16 :

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k}\Big(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6}\Big).$$

Calculer la n-ième décimale de π est équivalent à calculer la première décimale de la partie fractionnaire de $16^{n-1}\pi$. La formule BBP permet de calculer cette décimale sans avoir besoin de calculer toutes les décimales précédentes et d'une précision très grande.

Pour $j \geq 1$, posons

$$S(j) = \sum_{k=0}^{\infty} \frac{1}{16^k}\Big(\frac{1}{8k+j}\Big).$$

Pour calculer la n-ième décimale de π en base 16, nous devons calculer la première décimale de

$$S(j,n) = \{16^{n-1}S(j)\} = \left\{ \sum_{k=0}^{n-1} \frac{16^{n-1-k} \bmod (8k+j)}{8k+j} + \sum_{k=n}^{\infty} \frac{16^{n-1-k}}{8k+j} \right\}.$$

On évaluera séparément la somme de droite et celle de gauche.

Question à développer pendant l'oral : Montrer que $\sum_{k=n+15}^{\infty} \frac{16^{n-1-k}}{8k+j} \leq 1/2^{65}$, décrire l'algorithme pour calculer la fonction $S(j,n)$ et celui pour évaluer π en base 16, et donner leur complexité en temps.

Question 3 *Que valent :*
a) $S(1,u_0)$ ***b)*** $S(4,u_0)$ ***c)*** $S(5,u_2)$ ***d)*** $S(6,u_2)$.

Question 4 *La formule précédente permet d'obtenir plusieurs décimales de π commencant à la position n. Que valent les 4 décimales de π en base 16 commençant à la position (vous donnerez la valeur des décimales en base 10 et en base 16) :*
a) $n = u_0$ *?* ***b)*** $n = u_2$ *?* ***c)*** $n = u_{100}$ *?*

3 Décimales de π en base 10

On va utiliser une célèbre formule permettant de calculer en base 10

$$\frac{\pi}{4} = \arctan(1) = \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1}.$$

En utilisant une méthode pour accélérer la convergence d'une série alternée, on peut montrer qu'il existe deux entiers M et N tels que $|S_{i_0} - \pi| \leq \frac{\pi}{(2eM)^N}$ pour $i_0 = (M+1)N$,

$$S_{i_0} = \sum_{k=0}^{i_0-1} (-1)^k \frac{4}{2k+1} - \sum_{k=0}^{N-1} (-1)^k \frac{4s_k}{2^N(2MN+2k+1)} \text{ et } s_k = \sum_{j=0}^{k} \binom{N}{j}. \qquad (1)$$

Comme précédemment, on multiplie cette formule par 10^n pour obtenir la n-ième décimale en base 10, et pour obtenir une précision de n_0 chiffres, on impose $|10^n S_{i_0} - 10^n \pi| < 10^{-n_0}$ soit $\pi/(2eM)^N < 1/10^{n+n_0}$. Posons $N = \left\lceil (n + n_0 + 1)\frac{\log(10)}{\log(2eM)} \right\rceil$. Si $M \geq 4$, la partie fractionnaire de $10^n\pi$ peut être approximée avec une erreur inférieure à 10^{-n_0} par la partie fractionnaire de $B - C$ avec B et C les deux sommes de $10^n S_{i_0}$ dans la formule (1).

Question à développer pendant l'oral : Justifier qu'on peut calculer les parties fractionnaires de B et C en calculant les valeurs suivantes.

$$B = \sum_{k=0}^{i_0-1} (-1)^k \frac{4 \times 10^n}{2k+1}, \tag{2}$$

$$C = \sum_{k=0}^{N-1} (-1)^k \frac{5^{N-2} 10^{n-N+2} s_k \bmod (2MN + 2k + 1)}{2MN + 2k + 1}. \tag{3}$$

Pour calculer C, on a besoin de calculer $s_k = \sum_{j=0}^{k} \binom{N}{j} \bmod m$ pour différentes valeurs de N, k et du module m et c'est le but des questions 5 à 8. La question 9 demandera de calculer la partie fractionnaire de $B - C$.

3.1 Inverse modulaire

Dans cette partie, on aura besoin de calculer des inverses modulaires, c'est-à-dire l'entier $0 \leq x < m$ tel que $ax = 1 \bmod m$ pour un a et m donnés et sera noté $a^{-1} \bmod m$. Nous rappelons que cette fonction est bien définie si $\text{pgcd}(a, m) = 1$.
Le théorème de Bézout permet d'affirmer qu'il existe deux entiers x, y tels que

$$ax + my = \text{pgcd}(a, m).$$

Les coefficients de Bézout x et y se calculent aisément en utilisant l'algorithme d'Euclide étendu. L'idée consiste à maintenir l'invariant

$$ax_k + my_k = r_k,$$

où r_k est le reste de la k-ième division euclidienne de r_{k-2} par r_{k-1} et x_k et y_k les valeurs de x et y à cette itération. On initialisera les valeurs $x_0 = 1$, $x_1 = 0$, $y_0 = 0$ et $y_1 = 1$ de sorte que $r_0 = a$ et $r_1 = m$.

Question à développer pendant l'oral : Justifier que les suites $(x_k)_{k\geq 0}$ et $(y_k)_{k\geq 0}$ satisfont les relations de récurrence $x_{k+1} = x_{k-1} - x_k q_k$ et $y_{k+1} = y_{k-1} - y_k q_k$ où q_k est défini par $r_{k-2} = q_k r_{k-1} + r_k$ avec $0 \leq r_k < r_{k-1}$. Quelle est la complexité de votre algorithme ? En déduire comment calculer l'inverse modulaire à partir du calcul des coefficients de Bézout ?

Question 5 *Calculer* $(x, y, a^{-1} \bmod m)$ *pour les* a *et* m *suivants ou, le cas échéant, indiquer non inversible :*
a) $a = u_{10}, m = 647$ *?* ***b)*** $a = u_{15}, m = 12567$ ***c)*** $a = u_{100}, m = 120791$.

3.2 Factorisation

Proposer un algorithme permettant de calculer les décompositions en facteurs premiers de petits entiers.

Question 6 *Que valent les factorisations de :*
a) u_{10} ***b)*** u_{100} ***c)*** u_{1000}.

3.3 Calcul de somme de binomiaux modulaire

Pour calculer $s_k = \sum_{j=0}^{k} \binom{N}{j} \bmod m$, nous allons utiliser la formule

$$\binom{N}{j} = \frac{N-j+1}{j}\binom{N}{j-1}.$$

La difficulté vient du fait que le modulo m peut avoir des facteurs plus petits que j et on ne peut pas toujours inverser $j \bmod m$. Pour résoudre ce problème, nous calculons le pgcd de j et m à chaque étape de l'algorithme et on décomposera les binômes sous la forme

$$\binom{N}{j} = \frac{A}{B} \times R_{1,j} \times R_{2,j} \times \ldots \times R_{\ell,j}$$

avec A et B sans facteurs premiers $p \leq k$ de m, et chaque $R_{i,j}$ est une puissance du facteur premier p_i de m.
Question à développer pendant l'oral : À la j-ième étape, on pourra calculer trois entiers A_j, B_j et C_j tels que

$$\frac{\binom{N}{j}}{R_{1,j} \times R_{2,j} \times \ldots \times R_{\ell,j}} = \frac{A_j}{B_j} \text{ et } \sum_{i=0}^{j} \binom{N}{i} = \frac{C_j}{B_j} \bmod m.$$

Donner une relation de récurrence entre A_j, B_j, C_j, les facteurs de $N-j+1$ et de j, a^* et b^*, qui n'ont pas de facteurs communs avec les facteurs de m et les $R_{i,j}$s.

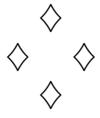
Question 7 *Que vaut* $s_k = \sum_{j=0}^{k} \binom{N}{j} \bmod m$ *pour les valeurs suivantes de* N, k, m *:*
a) $80, 10, u_{10}$ ***b)*** $400, 20, u_{100}$ ***c)*** $400, 100, u_{1000}$.

Question à développer pendant l'oral : Quelle est la complexité de l'algorithme ?

3.4 Calcul des décimales en base 10

On va choisir $M = 2\left\lceil \frac{n}{\log^3 n} \right\rceil$, où log représente le logarithme népérien.

Question 8 *Que valent les 4 décimales de* π *en base 10 en commençant à la* n*-ième avec une précision de* 10^{-4}.
a) $u_{10} \bmod 1000$ ***b)*** $u_{20} \bmod 2000$ ***c)*** $u_{30} \bmod 10000$.



Fiche réponse type : Décimales de π

$\widetilde{u_0}$: 1

Question 1

a) 3c.8b85

b) a.a1f8

c) 1000010001

d) 111000101

Question 2

a) 24657

b) 342

c) 1625

d) 33672

Question 3

a) 0.0072

b) 0.2554

c) 0.7338

d) 0.8526

Question 4

a) 0.1416, 0.243f

b) 0.3364, 0.561c

c) 0.3040, 0.4dd4

Question 5

a) -237, 22178, 410

b) 1840 -14091 non inv

c) 24377 -21458 24377

Question 6

a) $5^1 * 12109^1$

b) $13^1 * 8179^1$

c) $2^2 * 5^1 * 7^1 * 601^1$

Question 7

a) 24142

b) 105659

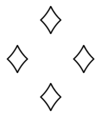
c) 66312

Question 8

a) 0.2179

b) 0.9860

c) 0.9042



Fiche réponse : Décimales de π

Nom, prénom, u_0 : .....

Question 1

a)

b)

c)

d)

Question 2

a)

b)

c)

d)

Question 3

a)

b)

c)

d)

Question 4

a)

b)

c)

Question 5

a)

b)

c)

Question 6

a)

b)

c)

Question 7

a)

b)

c)

Question 8

a)

b)

c)