G. Hunault
Licence Informatique

Angers, octobre 2000

Controle continu pour l'Option de Probabilités et Statistiques

Modalités du Controle

Le controle continu comprend deux petits projets à rendre, l'un concernant la partie Probabilités, l'autre la partie Statistiques. Chaque étudiant(e) doit rendre les deux projets. Chaque projet comporte une partie programmation et une partie rédaction qui comptent à part égale dans la note.

La rédaction se fait au choix mais exclusivement sous *Word* ou sous *Latex*. La programmation est effectuée au choix mais exclusivement en *Awk*, *Apl*, *Rexx*, *C*, *Perl*, *Tcl*, *Java* ou *Pascal*. On pourra ne rendre qu'un seul texte qui contiendra les algorithmes et explications des deux projets. Les documents (rédaction, algorithmes et programmes) seront transmis "de la main à la main" sous forme de fichier informatique sur disquette ou par *mail*. Ce fichier, unique, sera au format .zip ou .tar et les fichiers y seront archivés sans sous-répertoire. Ce fichier devra être remis au plus tard le

lundi 11 décembre, 12 h dernier délai

Pour éviter toute ambiguité, tous les noms de fichiers correspondront aux initiales de l'élève. Il sera possible de vérifier auprès de l'enseignant le 18 décembre entre 14 et 18 h que l'archive a bien été reçue et que les fichiers ont bien été décompressés et qu'ils sont lisibles, compilables etc.

1. Projet de Probabilités

La fonction factorielle $n \mapsto !n$ est utilisée pour de nombreux calculs de dénombrement. C'est une fonction qui à un nombre entier associe un nombre entier. On admettra démontré le résultat suivant : il existe un unique prolongement continu sur $\mathbb{R}$ de la fonction factorielle. Cette fonction, notée Γ est nommée fonction *Gamma*. Elle vérifie $\Gamma(n+1) = !n$ pour n entier.

Cherchez à la bibliothèque ou sur *Internet* une formule mathématique qui définit la fonction Γ ainsi qu'une méthode ou une formule pour simuler cette fonction. Vous rédigerez soigneusement la méthode, vous citerez vos sources bibliographiques et vous fournirez si possible un exemple, quitte à le recopier d'un ouvrage ou d'une page *Web* que vous citerez. Ecrire l'algorithme de la méthode retenue et traduisez-le dans un des langages suivants : *Awk*, *Apl*, *Rexx*, *C*, *Perl*, *Tcl*, *Java*, *Pascal*. Vous écrirez aussi un programme principal qui utilise cette approximation de la fonction Γ pour remplir un tableau construit sur le modèle suivant (chaque valeur sera affichée avec 2 décimales).

	0,0	0,1	0,2	...	0,8	0,9
1		0,95	0,92	...	0,93	0,96
2	1,00	1,05	1,10	...	1,68	1,83
3	2,00	2,20	2,42	...	4,69	5,30
4	6,00	6,81	7,76	...	17,84	20,67
5	24,00	27,93	32,58	...	85,62	101,27
6	120,00	142,45	169,41	...	496,61	597,49
7	720,00	868,96	1050,32	...	3376,92	4122,71
8	5040,00	6169,59	7562,29	...	26339,99	32569,40
9	40320,00	49973,71	62010,76	...	231791,88	289867,70
10	362880,00	454760,75	570499,03	...	2271560,42	2869690,27

Pour ceux et celles qui le désirent, on pourra aussi essayer de produire la suite du tableau, à savoir

	0,0	0,1	0,2	...	0,8	0,9
100	9.33e+155	1.48e+156	2.34e+156	...	3.71e+157	5.89e+157
110	1.44e+176	2.31e+176	3.69e+176	...	6.20e+177	9.92e+177
120	5.57e+196	8.99e+196	1.45e+197	...	2.57e+198	4.14e+198
...						
190	5.09e+349	8.61e+349	1.45e+350	...	3.39e+351	5.73e+351
200	3.94e+372	6.70e+372	1.14e+373	...	2.73e+374	4.64e+374

2. Projet de Statistiques

Soient X et Y deux variables logiques, c'est à dire deux colonnes indicées de chiffres 0 et 1. On dit que X et Y sont en contravis à l'indice i si la i-ème valeur de X, notée X_i est différente de la i-ème valeur de Y, notée Y_i. On appelle *taux de contravis* de X et de Y que l'on note $t(X, Y)$ le rapport du nombre de contravis sur le nombre total de valeurs. On appelle *matrice de contravis* des variables logiques V_i la matrice symétrique m telle que $m_{i,j} = t(V_i, V_j)$. Par exemple pour $V_1 = (1, 0, 0)$, $V_2 = (1, 1, 1)$ et $V_3 = (0, 0, 0)$ la matrice de contravis, que l'on donne sous forme triangulaire inférieure, vaut

```
0
0.67    0
0.33    1.00    0
```

On suppose désormais qu'il y a p variables de chacune n valeurs mises dans un fichier-texte, à raison de p valeurs par ligne, la ligne i du fichier contenant les valeurs à l'indice i des variables. Par exemple, on pourrait utiliser le fichier suivant de variables logiques :

```
1    0    0    0    0
1    0    1    1    1
1    0    1    1    1
1    0    1    1    1
1    1    1    1    1
1    1    1    1    1
1    1    1    1    1
```

Ecrire un algorithme complet (en-tête, paramètres, variables locales, variables globales et corps de l'algorithme) nommé `tauxContr` qui fournit la taux de contravis de deux variables logiques X et Y de n valeurs. Ensuite, si on suppose connues les fonctions algorithmiques `mots` et `mot` dont vous rappelerez la syntaxe, donner un algorithme qui fournit la matrice de contravis des variables à partir d'un fichier de variables logiques. (on utilisera la fonction `tauxContr` définie par l'algorithme précédent). Traduire en *Awk*, *Apl*, *Rexx*, *C*, *Perl*, *Tcl*, *Java* ou *Pascal* les algorithmes et en faire un programme complet. Le programme utilisera un et un seul paramètre, à savoir celui du nom du fichier qui contient les valeurs des variables.

Remarques concernant la rédaction

La note sera principalement fonction de la qualité de l'analyse du travail à effectuer, de l'exposé de la méthode et de la clarté de la rédaction. La valeur attribuée au programme ne représentera que 25 % de la note. Aucune interface n'est demandée, les programmes devront être exécutables en ligne de commande. En cas de langage compilé, on fournira la source et éventuellement l'exécutable produit. On précisera, si besoin est, la commande de compilation. Dans tous les cas de figures, on précisera la version du langage choisi, le type d'ordinateur et le système d'exploitation utilisés.

Aucun langage autre que cité ne devra être utilisé, aucun traitement de textes autre que cité ne devra être utilisé. Tout fichier qui ne correspondra pas aux initiales de l'élève ne sera pas lu. En particulier, les fichiers nommés proba, algo, stat,projet, doc ne seront pas lus.

Un programme même incomplet mais commenté et bien indenté sera au moins autant apprécié qu'un programme qui "marche" mais sans commentaire et illisible quand aux structures de flot (boucles, tests etc.). Les identificateurs auront tous au moins 4 caractères.

3. Références

Ce texte, de même que le cours de *Probabilités et Statistiques* pour la licence informatique et les T.D. correspondants, peut être récupéré à l'adresse

`http://www.info.univ-angers.fr/pub/gh/wstat/ghstat.htm`

Un manuel élémentaire d'*Algorithmique raisonnée* est sur la page

`http://www.info.univ-angers.fr/pub/gh/wals/alsgen.htm`

De plus, il est possible de tester si le projet a été correctement reçu (s'il a été envoyé avec le sujet "projliciy2k licence info" à gilles.hunaultuniv-angers.fr) grâce à la page

`http://www.info.univ-angers.fr/pub/gh/wstat/zipdirpli.php3`