Université de Paris 7 | **TP11 - Exemple Jury-1** | **agreg 2018 (F.Han [1])**

Exercice I: initiation à la syntaxe des corps non premier

Dans xcas ou sage, on peut créer des corps de cardinal non premier.

1) Créer un corps à 8 éléments, il sera noté `F8` et la variable d'extension sera notée `a`

2) Vérifiez que tout fonctionne comme vous le souhaitez en calculant `a+a` et `a^5`. Récupérez le polynôme minimal de `a` (on l'écrira comme un polynôme en la variable `x`)

3) Si la variable `P` contient l'élément $15 + x^3 + x^6$ de $\mathbb{Z}[x]$, factorisez `P` dans $\mathbb{F}_8[x]$.

Exercice II: Texte jury1 : Différentes façons de faire jouer le jury

1) Niveau 0 : On rentre la réponse à la main, ainsi que la matrice et on utilise un produit de matrice.

2) On améliore avec une procédure. (Comment faire apparaitre une fenêtre de saisie)

3) Améliorations : Création de H automatiquement.

4) Améliorations : On va expliquer maintenant comment sont crées les questions.

a) Création d'une bijection entre les mots du code et les entiers de 0 à 15.

b) Création des questions (ie des listes) automatiquement. On pourra dans un premier temps retourner des listes de vecteurs de $\mathbb{F}_2^7$.

Exercice III: Codes de Hamming

Il existe 2 façons de généraliser l'exemple précédent : Augmenter le nombre de lignes de H, et changer de corps. On notera $H_{r,q}$ une matrice de taille $r \times (q^r - 1)$ dont les colonnes sont paramétrées par les éléments de $\mathbb{P}(\mathbb{F}_q^r)$ en prennant un vecteur directeur de chaque droite vectorielle de $\mathbb{F}_q^r$. Le code correspondant sera encore le noyau de cette matrice.

1) Expliquer pourquoi on ne travaille pas modulo 4 ?

2) Comment généraliser l'exemple, étudier une généralisation sur $\mathbb{F}_4$? Indications :

a) Créer une liste `l` de tous les éléments de $\mathbb{F}_4$:

b) Créer tous les vecteurs de $\mathbb{F}_4^2$.

c) Donner un vecteur directeur de chaque droite vectorielle de $\mathbb{F}_4^2$ puis un vecteur directeur de chaque droite vectorielle de $\mathbb{F}_4^3$. On notera $H4$ la matrice 3×21 associée.

d) Trouver le noyau de $H4$.

Exercice IV:

Dans cas d'un code non linéaire sans règles particulières, on n'a pas d'autre moyen que d'utiliser les algorithmes de tri.

1) Créer une fonction distance de hamming entre 2 vecteurs.

2) On considère un code (non forcément linéaire) inclus dans $\mathbb{F}_3^{10}$ à 200 éléments.

a) Créer un tel code en tirant au hasard les coordonnées de ses éléments. On le représentera par une matrice 200 représentant la liste de tous les éléments du code. (Attention à ne pas confondre avec un code linéaire où la matrice

b) Etudier les fonctions de tri. Comment peut on en utiliser pour extraire d'une liste tous les éléments a distance 1 d'un vecteur donne ?

1. `http://webusers.imj-prg/frederic.han/agreg`

Chapitre 5

Épreuves orales de modélisation

Lors de l'inscription au concours, quatre options sont proposées :

A. Probabilités et Statistiques,
B. Calcul scientifique,
C. Algèbre et Calcul formel,
D. Modélisation et Analyse de systèmes informatiques.

L'épreuve de modélisation comporte une période de préparation de quatre heures et une interrogation dont la durée sera d'une heure à compter de 2017.

Même s'il s'agit d'une épreuve plus appliquée ou moins académique que les deux autres épreuves orales, cela ne dispense en aucun cas les candidats de faire preuve de la rigueur mathématique requise : quand on utilise un théorème, il faut impérativement être capable d'en restituer un jeu d'hypothèses. Par jeu d'hypothèses correct, on entend que le théorème soit vrai et qu'il s'applique effectivement au contexte considéré. Le jury n'attend pas nécessairement un énoncé avec les hypothèses minimales.

Cette épreuve de modélisation doit permettre aux candidats de mettre en avant diverses qualités : les connaissances mathématiques (pour les options A, B, C) / informatiques et mathématiques (pour l'option D), la réflexion et la mise en perspective des connaissances, l'aptitude à les appliquer à des problèmes concrets de modélisation et à produire des illustrations informatiques pertinentes, les qualités pédagogiques de mise en forme d'un exposé construit et cohérent, la capacité à faire preuve d'initiatives pour s'exprimer et manifester des qualités pédagogiques et de synthèse. La capacité des candidats à répondre aux questions fait partie intégrante de l'évaluation de cette épreuve. Comme pour l'ensemble des oraux, le caractère vivant de l'exposé est un atout.

Le texte fourni est la base pour construire et exposer un traitement mathématique (et/ou informatique pour l'option D) d'un problème « concret » en s'appuyant sur des éléments, généralement partiels, disséminés dans le texte. La présentation doit s'appuyer sur un dosage cohérent et harmonieux entre introduction motivée de modèles, preuves mathématiques, illustrations informatiques, critiques éventuelles du texte, réponses aux questions, mise en lumière des connaissances. Les candidats doivent utiliser leurs connaissances mathématiques et informatiques pour justifier certains points mentionnés dans le texte, proposer un retour sur le modèle ainsi qu'une conclusion.

Texte L'épreuve de modélisation repose sur un texte d'environ 5 à 6 pages (hors l'exercice de programmation pour l'option D) que les candidats choisissent parmi les deux qui leur sont proposés.

En 2017, ces textes seront surmontés, pour les options A, B, C, du bandeau suivant :

> *Il est rappelé que le jury n'exige pas une compréhension exhaustive du texte. La présentation, bien que totalement libre, doit être organisée et le jury apprécie qu'un plan soit*

annoncé en préliminaire. L'exposé doit être construit en évitant la paraphrase et en mettant en lumière les connaissances, à partir des éléments du texte. Il doit contenir des illustrations informatiques réalisées sur ordinateur, ou, à défaut, des propositions de telles illustrations. Des pistes de réflexion, indicatives et largement indépendantes les unes des autres, vous sont proposées en fin de texte.

Pour l'option D, le bandeau sera :

Il est rappelé que le jury n'exige pas une compréhension exhaustive du texte. La présentation, bien que totalement libre, doit être organisée et le jury apprécie qu'un plan soit annoncé en préliminaire. L'exposé doit être construit en évitant la paraphrase et en mettant en lumière les connaissances, à partir des éléments du texte. L'exercice de programmation doit être présenté lors de l'exposé. Des pistes de réflexion, indicatives et largement indépendantes les unes des autres, vous sont proposées en fin de texte.

Les textes se termineront par le texte suivant pour les options A, B, C :

Les pistes de réflexion suivantes ne sont qu'indicatives et il n'est pas obligatoire de les suivre. Vous pouvez choisir d'étudier, ou non, certains des points proposés, de façon plus ou moins approfondie, mais aussi toute autre question à votre initiative. Vos investigations comporteront une partie traitée sur ordinateur et, si possible, des représentations graphiques de vos résultats. À défaut, si vos illustrations informatiques n'ont pas abouti, il est conseillé d'expliquer ce que vous auriez souhaité mettre en œuvre.

Pour l'option D, la fin des textes comportera la mention suivante :

Les pistes de réflexion suivantes ne sont qu'indicatives et il n'est pas obligatoire de les suivre. Vous pouvez choisir d'étudier, ou non, certains des points proposés, de façon plus ou moins approfondie, mais aussi toute autre question à votre initiative. Vos investigations comporteront une partie traitée sur ordinateur, notamment à travers l'exercice de programmation.

Les textes sont souvent motivés par des problèmes concrets. Ils peuvent présenter des arguments rapides, voire heuristiques (signalés comme tels) mais ne contiennent pas d'assertion délibérément trompeuse et se concluent par une liste de suggestions. Même si la plupart des textes s'appuient sur des problématiques issues de sciences autres que les mathématiques, aucune connaissance dans ces domaines n'est exigée par le jury. Discuter la modélisation proposée par un texte consiste donc avant tout à dégager les comportements qualitatifs du modèle, la manière dont il dépend d'éventuels paramètres et s'il semble apte à rendre compte des phénomènes qu'il est censé représenter. Le jury s'attend à ce que les candidats ne se contentent pas d'un exposé qualitatif et démontrent certains résultats évoqués dans le texte. *A contrario*, les interprétations qualitatives du comportement des modèles sont parfois absentes des exposés. Pourtant, montrer que l'on comprend un modèle ne se réduit pas à prouver un théorème.

Chaque année, des textes sont rendus publics et sont disponibles sur le site de l'agrégation de mathématiques http://agreg.org. Ces textes sont représentatifs de l'épreuve et permettent aux candidats de se familiariser avec le format des textes, se faire une idée des attentes du jury, réfléchir à des illustrations numériques pertinentes dans le cadre du texte ou avoir des exemples d'exercices de programmation pour l'option D.

Préparation Durant les quatre heures de préparation, les candidats ont accès à la bibliothèque et disposent d'un ordinateur dont la configuration est décrite sur le site de l'agrégation de mathématiques, à l'adresse http://agreg.org. Les candidats retrouveront le même environnement sur l'ordinateur de la salle d'interrogation. Il n'est évidemment pas réaliste de découvrir les logiciels à disposition des candidats le jour de l'épreuve : la configuration informatique utilisée pour le concours et sa documentation

sont accessibles et téléchargeables sur le site http://clefagreg.dnsalias.org/ et permettent de se familiariser avec l'environnement offert pour l'épreuve.

Il est recommandé aux candidats de consacrer une partie de leur temps de préparation à s'interroger sur le discours qu'ils tiendront, la stratégie d'exploitation du tableau et d'utilisation de l'outil informatique qui leur permettra de mettre au mieux en valeur leurs connaissances et leur compréhension du texte ou d'une partie de celui-ci. Proposer un exposé structuré et cohérent ne peut s'improviser au moment de l'oral et doit faire l'objet d'une réflexion préalable durant la préparation.

Oral Au début de l'interrogation, le jury commence par vérifier que les fichiers créés par les candidats lors de la préparation ont bien été transférés sur la machine locale (dont l'environnement est identifique à celui de la salle de préparation) et par rappeler les modalités de l'épreuve. Celle-ci est scindée en deux temps : un exposé de 40 minutes (35 minutes à partir de 2017), suivi de 25 minutes d'échanges avec le jury. Les candidats sont invités à commencer par donner la structure de leur présentation sous forme d'un plan et le déroulement de l'exposé doit être en cohérence avec cette structure. Grâce à ce plan, le jury pourra ainsi avoir une vision globale de l'exposé et aider, si besoin, les candidats à gérer leur temps. Comme les candidats se le voient rappeler en début d'épreuve, l'exposé doit être accessible à un public qui découvre les problématiques du texte et doit permettre d'en faire comprendre les enjeux à un public qui ne le connaîtrait pas. Le jury, tout en étant conscient des difficultés du concours, attend un minimum d'aisance au tableau, la manifestation d'une certaine volonté de capter l'attention de l'auditoire et un discours clair et précis.

Durant l'exposé, les candidats disposent de leurs notes, d'un tableau et d'un ordinateur. Ils peuvent alterner, quand bon leur semble, entre un exposé oral, quelques éléments rédigés au tableau de façon propre et lisible et la présentation de ce qui a été préparé à l'aide de l'outil informatique. Les candidats doivent gérer correctement le tableau et demander, si besoin, au jury les parties qu'ils peuvent effacer (le jury peut souhaiter conserver certains passages et y revenir lors des échanges avec les candidats). Le jury a sous les yeux un exemplaire des textes et les candidats peuvent y faire référence pour éviter de recopier les notations, les énoncés complets ou certaines formules. Même si les programmes ne fonctionnent pas comme ils l'auraient souhaité ou si les simulations numériques n'ont pas abouti, les candidats sont invités à expliquer ce qu'ils voulaient mettre en œuvre, illustrer ou programmer. Si les candidats n'ont pas accédé à l'ordinateur dix minutes avant la fin du temps qui leur est imparti, le jury les préviendra.

Il n'y a pas de « format type » pour cette épreuve. Des prestations très différentes, tant dans leur forme que dans leur contenu, sur un même texte, peuvent conduire également à des notes élevées. Comme dans tout oral, la construction de l'exposé doit être une préoccupation importante des candidats. Une réflexion s'impose afin de produire un tout cohérent et intelligible par un public qui, dans l'esprit de l'épreuve, découvre le texte à travers l'exposé des candidats. Une brève introduction à la problématique avant de s'engager dans une longue digression (succession de définitions, théorèmes, ...) sans lien avec le problème de départ ne peut donner satisfaction. Il en va de même d'un exposé se réduisant à la présentation de la problématique du texte et à des illustrations informatiques, ou à l'énumération linéaire des pistes de réflexion proposées par le texte, sans contribution des candidats. Un texte traité de façon partielle mais en profondeur peut donner une note élevée. À un survol superficiel de l'intégralité du texte sans apport mathématique / informatique ou critique scientifique, les candidats doivent préférer une discussion fouillée d'une portion du texte, bâtie sur des arguments mathématiques / informatiques solides, des simulations pertinentes accompagnées de commentaires de bon aloi. Néanmoins, dans l'ensemble, les candidats semblent avoir perçu la nécessité d'utiliser, au mieux, le temps qui leur est consacré. Le jury apprécie de voir de plus en plus de candidats qui se sont approprié le texte et en donnent une présentation pertinente et autre qu'une paraphrase linéaire.

Pour enrichir leur propos, les candidats sont invités à mobiliser leurs connaissances, sur des aspects variés du programme, en étayant les arguments seulement esquissés dans le texte par des énoncés précis.

Il est totalement illusoire de chercher à impressionner le jury par une logorrhée de mots savants : les textes proposés peuvent être discutés en exploitant un bagage technique qui n'utilise pas les éléments les plus sophistiqués du programme. En particulier, le jury ne manque pas de s'attarder sur toute notion amenée par les candidats durant leur présentation et il est toujours dommageable de s'aventurer sur des terrains méconnus. Bien plus qu'une démonstration de virtuosité technique, le jury attend que les candidats montrent leur maîtrise d'énoncés relativement simples « en situation » : c'est là que réside une des difficultés principales de l'épreuve. Nombre de candidats éprouvent des difficultés à formaliser précisément des notions de base du programme ou à mettre en œuvre certaines de leurs connaissances. *A contrario*, utiliser une portion excessive du temps de parole pour recycler un chapitre de cours ou un développement d'une leçon d'Analyse et Probabilités, d'Algèbre et Géométrie ou d'Informatique, en s'éloignant des enjeux du texte est considéré comme un hors sujet et est sévèrement sanctionné. La paraphrase pure et simple d'amples portions du texte ne constitue en aucun cas un exposé satisfaisant. Les textes fournissent souvent des esquisses de démonstrations qui sont précisément destinées à être complétées et commentées. Les candidats ne doivent pas se contenter de ces esquisses de démonstration. S'ils font mention de ces esquisses, le jury s'assurera que les candidats ont compris en profondeur. Le jury n'est pas dupe des candidats qui tentent de faire semblant de connaître une notion ou d'avoir compris un point du texte ou une démonstration. Il ne se laisse pas tromper non plus par les candidats qui font des indications du texte un argument d'autorité, tentative maladroite de masquer des insuffisances techniques. Un regard critique (« il faudrait prouver que... mais je n'ai pas réussi à le faire », « les hypothèses du théorème de XXX que je connais pour aborder des problèmes similaires ne sont pas satisfaites dans le cas présent » ...) est une attitude bien plus payante.

Échanges avec le jury Durant 25 minutes, le jury revient sur certains propos des candidats qui méritent précision. Il peut demander l'énoncé précis de théorèmes utilisés pour démontrer une assertion. Les échanges peuvent également porter sur la modélisation, les illustrations informatiques ou les exercices de programmation pour l'option D.

5.1 Recommandations du jury, communes aux options A, B, C

Le jury tient à souligner les attentes partagées entre les trois options A (Probabilités et Statistiques), B (Calcul scientifique) et C (Algèbre et Calcul formel).

Le jury regrette de ne pas voir davantage de dessins (soignés) ou schémas explicatifs qui peuvent rendre l'argumentation plus claire et convaincante. La capacité à revenir sur le problème de départ et à conclure quant à l'efficacité de l'approche mathématique proposée pour y répondre est une qualité très appréciée. La rigueur et la clarté de l'organisation, la gestion du temps, la pertinence des choix opérés parmi les différentes questions soulevées par le texte sont des éléments de l'évaluation. Les qualités de synthèse sont aussi appelées à s'exprimer : il vaut mieux indiquer les étapes cruciales d'un raisonnement que de se lancer dans un long calcul fastidieux qu'on aura du mal à mener à bien.

5.1.1 Illustration informatique

Le jury rappelle son fort attachement à cet aspect de l'épreuve, dont les ambitions sont clairement délimitées. Il ne s'agit en aucun cas, et pour aucune de ces trois options, d'un exercice de programmation. L'objectif est d'être capable d'utiliser l'outil informatique pour illustrer, de façon pertinente, le contenu du texte. La réalisation de cet objectif constitue une part incompressible de la note finale attribuée à l'épreuve. Une très bonne évaluation peut résulter d'une exploitation judicieuse de programmes simples, reposant largement sur les routines standards des logiciels fournis. La forme et la nature des illustrations proposées n'obéissent à aucun format préétabli. En revanche, elles doivent faire la preuve d'une véritable réflexion scientifique et être agrémentées de commentaires, sur les résultats et

les méthodes. Les illustrations informatiques devraient plus souvent permettre de faire un retour sur la problématique du texte et il convient de commenter les résultats obtenus dans le cadre du modèle proposé par le texte. Des illustrations peuvent mettre en valeur les limites du modèle étudié et le jury apprécie particulièrement que les candidats aient un regard critique sur le texte et ne prennent pas celui-ci pour vérité absolue. Le jury rappelle que même si les simulations ne sont pas abouties, il sait valoriser la démarche suivie lorsqu'elle est clairement argumentée et permettrait, avec des aménagements mineurs, de mettre en évidence des aspects intéressants du texte.

Si un bel effort est produit par la majorité des candidats, un nombre limité de candidats préfèrent ne pas toucher à l'outil informatique. Si les candidats pensent ainsi consacrer plus de temps à l'analyse du texte et à la préparation de leur exposé oral, cette stratégie n'est en aucun cas payante, bien au contraire.

Il est important de se rappeler qu'il s'agit d'une illustration. Présenter le code source de façon détaillée n'est pas indispensable, le jury sera plus sensible à une présentation de résultats d'exécution et/ou une représentation graphique clairement commentées oralement par le candidat. Dans le cas d'une représentation graphique, il est important de préciser, au moins oralement, ce qui est représenté en abscisses et ordonnées.

5.1.2 Liens avec les épreuves d'Analyse-Probabilités et Algèbre-Géométrie

Les candidats ne doivent pas hésiter à s'appuyer sur leurs connaissances spécifiques pour proposer des développements originaux dans les leçons d'Analyse et Probabilités ou d'Algèbre et Géométrie. Par exemple,

A. les candidats de l'option A peuvent nourrir leurs leçons de thèmes et d'exemples probabilistes (la fonction caractéristique d'une variable aléatoire est une transformée de Fourier, le produit de convolution a un sens probabiliste, les espaces L^p d'une mesure de probabilité sont intéressants, l'étude de certaines séries aléatoires est possible, la fonction de répartition est une fonction monotone digne d'intérêt, le calcul intégral est en lien étroit avec les probabilités, ...).

B. les candidats de l'option B peuvent centrer leur propos sur des problématiques motivées par les préoccupations du calcul scientifique (approximation de solutions d'équations linéaires ou non linéaires, d'équations différentielles, convergence d'algorithmes, analyse matricielle,...).

C. les candidats de l'option C peuvent s'appuyer sur leurs connaissances spécifiques en algèbre effective.

5.2 Option A : Probabilités et Statistiques

5.2.1 Commentaires généraux

Les deux aspects, probabiliste et statistique, forment un tout cohérent dans l'étude des phénomènes aléatoires et les textes proposés mêlent souvent ces deux points de vue. Cependant, il n'est pas impossible de se voir proposer le choix entre deux textes où l'aspect statistique est plus marqué : il est donc nécessaire que les candidats fassent un effort de formation statistique. La part importante de la statistique dans l'enseignement des mathématiques justifie cet investissement. Ainsi, le jury se satisfait de ne plus voir les candidats bouder les textes à coloration statistique, contrairement à ce qui a pu être constaté lors des sessions précédentes.

En outre, la difficulté des textes étant progressive, une honnête maîtrise des fondamentaux du programme permet, sans virtuosité technique, d'aborder l'épreuve favorablement. Il n'est pas rare qu'un texte démarre avec des résultats classiques (par exemple, la loi des grands nombres ou le théorème central limite). Être à l'aise avec ces notions permet de démarrer l'exposé en confiance.

— **Optimisation.** Les théorèmes d'existence d'optima sont mal connus. On s'attend à ce que les candidats soient capable de préciser les hypothèses requises pour :
  1. obtenir l'existence d'un optimum,
  2. obtenir l'unicité,
  3. caractériser le minimum (sans contrainte ou avec contraintes d'égalité).

— **Analyse de Fourier.** Le jury s'étonne que de nombreux candidats aient du mal à préciser les hypothèses de régularité assurant la convergence de la série de Fourier. Le lien entre la régularité de la fonction et le comportement asymptotique de ses coefficients de Fourier doit être connu.

5.3.3 Modélisation et mise en œuvre informatique

Le jury de l'option B rappelle qu'une illustration réalisée avec les routines de base des logiciels fournis est tout à fait satisfaisante si elle est clairement présentée, motivée et discutée. Si `Scilab` ou `Python` sont certainement les logiciels les mieux adaptés, le jury relève qu'un certain nombre de candidats a pu fournir des résultats convaincants avec un logiciel comme `Octave`, `XCas` ou `Sage`. Le jury insiste sur le fait qu'il est important de commenter les résultats informatiques (courbes, solutions, erreurs, ...) obtenus et de les relier au problème de modélisation traité par le texte ou à des résultats mathématiques sous-jacents.

5.4 Option C : Algèbre et Calcul formel

5.4.1 Commentaires généraux

La ligne directrice de l'option C est dans un premier temps la recherche de l'*effectivité*, puis de l'*efficacité* (souvent en temps, parfois en espace) du calcul algébrique. Cela va des aspects les plus élémentaires (algèbre linéaire, groupes, calcul avec les entiers, les polynômes, les entiers modulo) aux aspects plus avancés (élimination, géométrie effective, codes correcteurs d'erreur). La quasi-totalité des textes proposés dans le cadre de l'option rentrent totalement dans une ou plusieurs de ces grandes problématiques.

Cette « grille de lecture » peut accompagner les candidats, les aider à construire leur exposé et, dans l'idéal, les amener à expliquer non seulement *ce que* le texte fait, mais aussi *pourquoi* il le fait. La capacité à percevoir ces problématiques fait la différence entre bonnes et excellentes prestations, et peut aussi expliquer les notes honorables de certains candidats dont le niveau mathématique était pourtant limité.

5.4.2 Calcul algébrique effectif

Lorsqu'un énoncé du texte affirme l'existence d'un objet (scalaire, vecteur, matrice, élément d'un groupe, entier, polynôme, ...), le jury apprécierait que les candidats mènent la réflexion suivante :
— peut-on calculer cet objet ?
— si oui, par quelle(s) méthode(s) ?
— ces méthodes sont-elles efficaces ? quel est leur coût ?
Les candidats ayant le réflexe de se saisir, spontanément, d'une question de complexité, sont encore très rares. Pourtant cette démarche est perçue très positivement par le jury. Pour prendre un exemple, quand le texte parle de cryptographie, comparer le coût du système proposé et le coût d'une attaque, même naïve, est une initiative intéressante et actuellement quasi–inexistante. Une telle étude est beaucoup plus à sa place qu'un exposé détaillé de RSA plaqué sur un texte qui n'en parle pas – la mention rapide de RSA dans un texte introduisant un système de chiffrement pour comparer des complexités restant bien sûr pertinente. Plus largement, une réflexion minimale sur les ordres de grandeur (est-ce

qu'un calcul faisable représente $10^1, 10^{10}, 10^{100}, 10^{1000}$ opérations élémentaires ?) permettrait souvent de mieux situer les problèmes soulevés par un texte, ou de proposer des valeurs de paramètres réalistes quand ce sujet n'est pas évoqué par le texte.

5.4.3 Modélisation et mise en œuvre informatique

Le jury constate avec satisfaction une progression de la moyenne des candidats dans l'utilisation de l'outil informatique et regrette parfois que des candidats ayant préparé un travail d'illustration raisonnable, ne le mettent pas plus en valeur durant l'exposé.

Rappelons cependant que les candidats doivent être capables de justifier la pertinence de leur programme ou de leurs calculs dans le cadre du texte. Reprendre un morceau de code d'un livre est une démarche tout-à-fait acceptable à condition que les candidats comprennent exactement ce que fait le code et que cela fasse sens dans le cadre du texte.

Pour finir, le jury est parfois surpris de voir des candidats développer de longs et fastidieux calculs au tableau alors que l'utilisation de l'outil informatique leur aurait permis de gagner en temps et en clarté.

5.4.4 Aspects mathématiques

Sur les aspects mathématiques et la maîtrise des éléments du programme, les principales observations du jury sont les suivantes :

— Par rapport aux années précédentes, le jury se réjouit d'observer une amélioration en ce qui concerne l'algèbre linéaire effective. Les applications de la méthode du pivot de Gauss sont un peu mieux maîtrisées par les candidats. De plus, la complexité de cet algorithme est maintenant fréquemment connue des candidats même si l'origine de ce $O(n^3)$ donne parfois encore lieu à des explications pour le moins obscures.

— Le jury constate également une progression dans les connaissances théoriques des candidats sur les corps finis. Malgré cela, le calcul effectif dans ces corps n'est pas toujours maîtrisé.

— *A contrario*, la connaissance du résultant semble avoir régressé par rapport aux sessions précédentes. Si les candidats interrogés sur le sujet sont en général capables d'en donner une définition, ses propriétés élémentaires et surtout son utilisation pour éliminer des variables dans un système d'équations polynomiales semble très floue pour de nombreux candidats. Beaucoup de candidats le voient comme un critère d'existence d'un facteur commun et ne pensent plus (surtout dans le cas des polynômes à une variable) au PGCD qui est un objet bien plus simple à appréhender.

— Le jury observe que de nombreux candidats n'ont aucune connaissance sur les codes correcteurs d'erreurs. Un bon nombre d'entre eux font d'ailleurs la confusion entre codes correcteurs d'erreurs et cryptographie. Rappelons que les codes correcteurs sont une partie limitée du programme, et que très peu de connaissances sont exigibles (et exigées). Toutefois, il est nécessaire de s'y être confronté pour se familiariser avec les problématiques sous-jacentes, à savoir, typiquement qu'un bon code correcteur se décrit de façon compacte (et est donc en général linéaire), a une grande dimension et grande distance minimale (par rapport à sa longueur) et, aussi et surtout un algorithme de décodage efficace – rappelons que ce second point n'est pas vrai d'un code linéaire « quelconque ». Il faut avoir déjà un peu étudié le sujet pour comprendre les questions soulevées par presque tout texte sur les codes. Signalons enfin que la méconnaissance des corps finis est souvent rédhibitoire pour ce sujet.

— Si l'algorithme d'Euclide est bien connu des candidats, la plupart d'entre eux ne savent obtenir des relations de Bézout qu'en effectuant l'algorithme d'Euclide classique puis en procédant à une laborieuse « remontée ». Rappelons que l'algorithme d'Euclide **étendu** est explicitement au programme de l'option.

— Les attentes du jury en termes de complexité sont limitées mais il est attendu des candidats qu'ils sachent estimer le coût de certaines procédures classiques au programme : opérations sur les entiers et dans $\mathbf{Z}/n\mathbf{Z}$, sur les polynômes (multiplication, division euclidienne, évaluation, interpolation), pivot de Gauss, algorithme d'Euclide, ...

5.5 Option D : Modélisation et Analyse de Systèmes Informatiques

5.5.1 Commentaires généraux

Le jury apprécie le travail accompli pour la préparation de cette épreuve par les meilleurs candidats. Les candidats sont interrogés dans le même esprit que dans les autres options et les critères d'évaluation sont largement semblables, sauf en ce qui concerne l'exercice de programmation.

Dans la suite, nous procéderons en deux temps : nous évoquerons d'abord la vision que le jury a du déroulement de l'épreuve, avant de revenir sur les travers les plus fréquents. Chacune de ces deux parties évoquera les différents temps de l'épreuve : exposé, exercice de programmation, questions.

5.5.2 Attentes du jury

Les textes présentent généralement une problématique concrète, informatique ou de la vie de tous les jours, avant d'en proposer une formalisation plus ou moins complète et une analyse informatique plus ou moins détaillée. Ils sont souvent plutôt de nature descriptive et volontairement allusifs.

Exposé du candidat

Motivation. C'est aux candidats d'introduire le sujet du texte et de motiver la présentation qui va suivre – et cela n'arrive, hélas, que très rarement. Cette motivation est le plus souvent l'évocation de situations concrètes dans lesquelles on a besoin d'outils informatiques spécifiques. Ces situations peuvent être proposées par le texte lui-même, mais elles peuvent aussi être tirées de l'expérience personnelle des candidats. Toute contribution personnelle à ce niveau est toujours très appréciée !

Présentation du texte. Il est attendu des candidats une restitution argumentée d'une partie cohérente du texte, ainsi qu'un effort de formalisation sur les parties descriptives et allusives du texte. Il est bon d'essayer de donner une ou des preuves complètes d'énoncés du texte, ou de compléter les arguments parfois lapidaires fournis par ce dernier. Les énoncés considérés comme vraiment trop difficiles pour être prouvés dans le cadre d'une préparation en temps limité en partant des connaissances du programme sont systématiquement pointés comme devant être admis.
Il est attendu des candidats qu'ils soient fidèles à l'esprit du texte. Il ne s'agit pas de traiter l'intégralité des points du texte, mais que le traitement choisi soit cohérent : les candidats doivent par exemple pouvoir expliquer pourquoi ils ont choisi de développer certains points, et pas certains autres.

Exercice de programmation informatique

Nature de l'exercice. L'exercice de programmation proposé est en règle générale très simple et peut presque toujours être traité en une vingtaine de lignes. Il est décrit par une spécification claire (entrée / sortie / hypothèses éventuelles sur les données) qui doit impérativement être respectée [1].
La simplicité de l'exercice vient du fait que le jury souhaite avant tout tester une capacité (et non une virtuosité) à organiser un programme simple, clair, et pédagogique : le programme doit pouvoir être présenté et les choix (structures de données, style impératif vs fonctionnel, types), argumentés.

1. à une réserve près sur liste/tableau évoquée *infra*