

Webinaire

Réunion sur l'agrégation externe d'informatique

Sylvie Boldo
présidente de l'agrégation externe d'informatique

2 décembre 2022

<https://agreg-info.org/>

- 1 Généralités
- 2 Épreuves écrites 2022
- 3 Épreuves orales 2022
- 4 Questions

Notes :

- Les slides seront disponibles.
- Vous pouvez utiliser l'onglet « Questions » pour poser ou voter pour des questions.

1 Généralités

2 Épreuves écrites 2022

3 Épreuves orales 2022

4 Questions

Quelques liens utiles

- <https://www.devenirenseignant.gouv.fr/>
(fait foi)
- <https://agreg-info.org/>
(sujets 2022, rapport 2022, informations 2023, FAQ...)
- arrêté MENH2112666A du 17 mai 2021
<https://www.legifrance.gouv.fr/jorf/id/JORFTEXT000043648279>

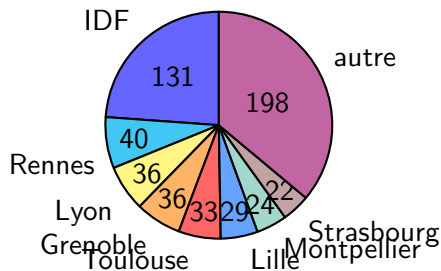
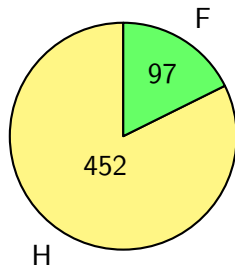
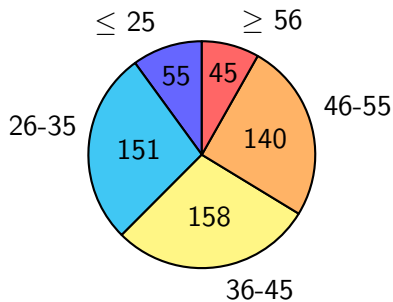
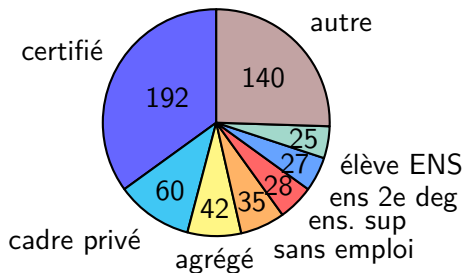
(+ liste de diffusion par email)

Inscrits	Présents	Admissibles	Admis & LC
549	251	55	23

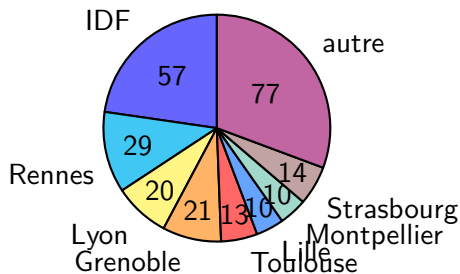
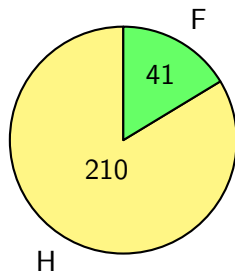
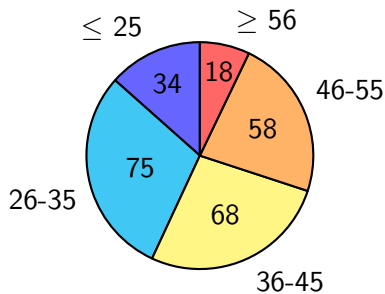
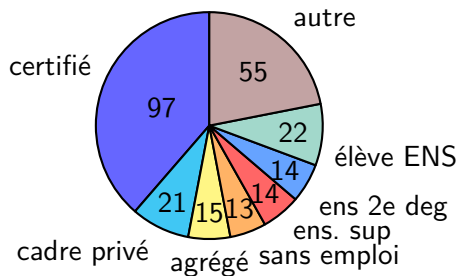
Barres :

- admissibilité : 10,47/20,
- admission : 12,86 pour la liste principale et
12,58 sur la liste complémentaire.

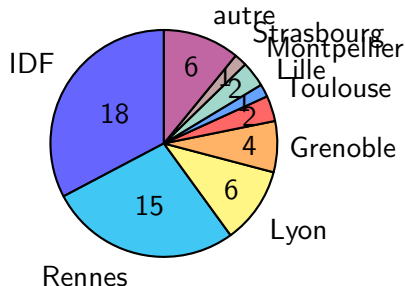
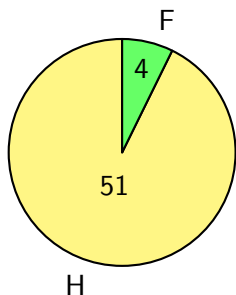
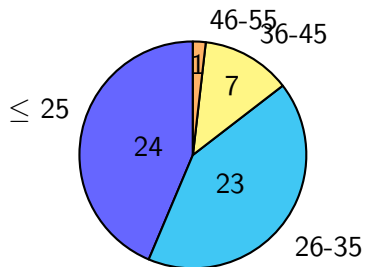
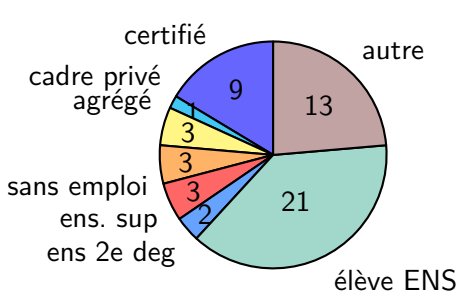
Inscrits (549)



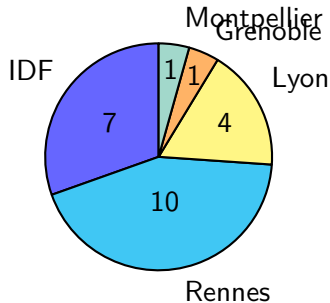
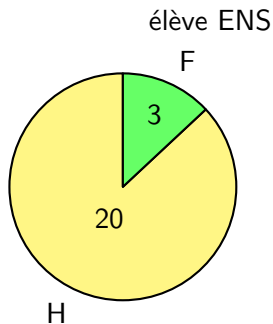
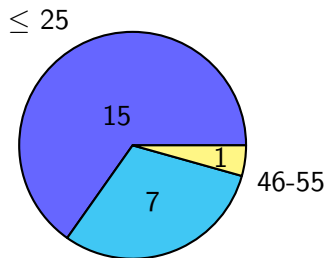
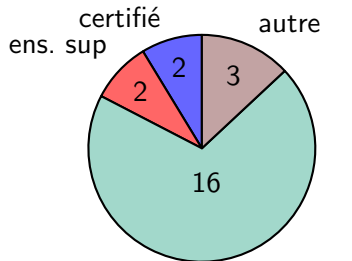
Présents (251)



Admissibles (55)



Admis & LC (23)



Éléments factuels sur les épreuves

- 3 épreuves écrites d'admissibilité (cette session : 6–8 mars 2023)
 - Composition d'informatique 5h (plusieurs problèmes ou exercices permettant d'en parcourir les grands domaines)
 - Étude d'un problème informatique 6h
 - au choix
 - Étude de cas informatique
 - Fondements de l'informatique
- 3 épreuves orales d'admission
 - Leçon d'informatique 1h (4h de préparation)
 - Modélisation 1h (4h de préparation)
 - Travaux pratiques de programmation 1h (5h de préparation)

1 Généralités

2 Épreuves écrites 2022

3 Épreuves orales 2022

4 Questions

Remarques générales

Points forts :

- copies globalement très satisfaisantes,
 - de très belles copies sur les épreuves 2 et 3.
- ⇒ On cherche des candidats solides et précis sur les bases de la discipline, afin de pouvoir transmettre ces dernières.

Points faibles :

- Attention à la rédaction et à la maîtrise de la langue.
- Ne pas proposer plusieurs versions de la réponse à une même question.

Épreuve 1 : composition d'informatique 1/2

L'épreuve vise à s'assurer de la maîtrise des concepts et des méthodes de la science informatique. Elle consiste en la résolution de plusieurs problèmes ou exercices permettant d'en parcourir les grands domaines.

Quatre exercices :

- système
- nombres flottants
- logique et déduction naturelle
- réseaux

Épreuve 1 : composition d'informatique 2/2

- 259 présents
- Meilleure note : 14,84/20
- Moyenne : 6,37 ; écart-type : 2,86.

$\geq 4,85$	75%
$\geq 5,00$	64%
$\geq 6,00$	50%
$\geq 8,30$	25%
$\geq 10,00$	12%
$\geq 15,00$	0%

- Erreurs de calcul manifestes.
- Peu de copies traitent tous les exercices.
- Déduction naturelle et arbres de preuve peu maîtrisés.
- Faiblesses en réseaux.

Le jury souhaite recruter des informaticiens **complets**.

Épreuve 2 : étude d'un problème informatique 1/2

L'épreuve vise à s'assurer de la capacité à se confronter à un problème informatique relativement ouvert et à prendre des initiatives pertinentes. Le sujet se compose de la présentation d'un problème et de questions permettant de guider le candidat.

On veut évaluer :

- algorithmique et programmation en Python,
- complexité, correction.

Thématique : construction d'une bibliothèque de grands entiers.

Épreuve 2 : étude d'un problème informatique 2/2

- 256 présents
- Meilleure note : 19,06/20
- Moyenne : 6,11 ; écart-type : 5,02.

$\geq 1,70$	75%
$\geq 5,00$	50%
$\geq 9,00$	25%
$\geq 10,00$	22%
$\geq 15,00$	7%

- De très belles copies.
- Faiblesses sur la preuve par induction et les preuves de correction.
- Faiblesses sur la complexité (rechercher dans une liste n'est pas $O(1)$)

Épreuve 3A : épreuve spécifique – étude de cas informatique 1/2

À partir d'un dossier présentant le cahier des charges de développement d'une application, le traitement du sujet doit permettre de vérifier la capacité du candidat à maîtriser l'ensemble des problématiques associées à l'application pouvant aller de l'analyse à la conception, du traitement des données et l'étude des infrastructures attenantes.

On veut évaluer :

- programmation et génie logiciel en Python,
- développement web et JavaScript,
- bases de données, architecture.

Thématique : problèmes de jeu d'échec.

Épreuve 3A : épreuve spécifique – étude de cas informatique 2/2

- 150 présents
- Meilleure note : 19,12/20
- Moyenne : 6,14, écart-type : 3,48

$\geq 4,25$	75%
$\geq 5,00$	67%
$\geq 6,10$	50%
$\geq 8,20$	25%
$\geq 10,00$	15%
$\geq 15,00$	3%

- De très belles copies.
- Niveau en programmation très hétérogène.
- Faiblesses en développement web.

Épreuve 3B : épreuve spécifique – fondements de l'informatique 1/2

Le traitement du sujet doit permettre de vérifier, au travers de la résolution d'un problème et/ ou de questions spécifiques, la maîtrise de concepts avancés d'informatique fondamentale et la capacité à les manipuler rigoureusement, à les relier entre eux et connaître leurs applications à des problématiques concrètes.

On veut évaluer :

- algorithmique et programmation OCaml,
- sémantique,
- jeu sur un graphe.

Thématique : compilation des expressions arithmétiques en tenant compte des besoins en espace et en temps.

Épreuve 3B : épreuve spécifique – fondements de l'informatique 2/2

- 104 présents
- Meilleure note : 19,24/20
- Moyenne : 9,32, écart-type : 5,35.

$\geq 4,75$	75%
$\geq 5,00$	72%
$\geq 8,00$	50%
$\geq 10,00$	48%
$\geq 14,40$	25%
$\geq 15,00$	20%

- De très belles copies.
- Beaucoup de bonnes copies.
- Parfois familiarité faible avec OCaml.
- Parfois faiblesses sur la preuve par induction.

1 Généralités

2 Épreuves écrites 2022

3 Épreuves orales 2022

4 Questions

Épreuves orales du 16 au 24 juin 2022 au lycée Paul Valéry à Paris.

- Les livres (de l'agrégation et des préparations) ainsi que l'environnement informatique sont disponibles pour toutes les épreuves.
- Les brouillons sont utilisables pendant les interrogations.
- Nourriture et boisson sont autorisées pendant la préparation.
(exemple pour leçon : convocation 7h ; interrogation 11h30-12h30)
(exemple pour TP : convocation 9h30 ; interrogation 15h-16h)

Points forts :

- très solide niveau global,
- beaucoup d'excellentes prestations, peu d'oraux vraiment ratés,
- à la fois sur le **disciplinaire** et la **pédagogie**.

Points faibles :

- Attention à la posture de professeur.
- Attention à la gestion du tableau.

Leçon (coefficient 2) 1/2

L'épreuve vise à s'assurer de la capacité du candidat à mobiliser et organiser des connaissances sur un thème donné, à les présenter clairement et rigoureusement et à interagir avec le jury. Elle consiste en la présentation d'une leçon, suivie d'un entretien avec le jury.

La liste des leçons pour 2023 est disponible. Elle a un peu changé.

Le candidat tire 2 leçons et en présente une (choix à l'entrée dans la salle).

Le jury attend un plan de la leçon d'au plus trois feuillets A4 manuscrits incluant deux développements.

Structure de l'oral : 10' de présentation du plan ; 20' de développement ; 30' de questions.

Leçon (coefficient 2) 2/2

- Présents : 54
- Meilleure note : 20
- Moyenne : 10,93, écart-type : 5,51

$\geq 5,00$	80%
$\geq 7,25$	75%
$\geq 10,00$	54%
$\geq 11,00$	50%
$\geq 15,00$	25%

- De très belles prestations.
- Souvent des plans structurés et détaillés, incluant des définitions, des exemples, des remarques, des illustrations...
- Une expertise sur des domaines hors programme ne se substitue pas à un recul pédagogique.

Modélisation 1/2

L'épreuve vise à s'assurer de la capacité du candidat à comprendre, critiquer et améliorer la modélisation informatique d'un problème éventuellement issu d'une autre discipline, à en exposer clairement les enjeux scientifiques, y compris dans leurs dimensions éthiques, sociétales, environnementales, économiques, ou encore juridiques.

Le candidat reçoit un sujet (exemples sur <https://agreg-info.org/>).

Structure de l'oral : 35' de présentation ; 25' de questions.

Le jury attend

- une présentation et discussion du problème et de sa formalisation,
- une capacité à développer, compléter, voire améliorer (ou critiquer) les ébauches de solution esquissées,
- une **illustration informatique**,
- une **discussion** d'une dimension éthique, sociétale, environnementale, économique ou juridique.

Modélisation 2/2

- Présents : 54
- Meilleure note : 20
- Moyenne : 10,13, écart-type : 4,39

$\geq 5,00$	81%
$\geq 6,50$	75%
$\geq 10,00$	56%
$\geq 11,00$	50%
$\geq 13,50$	25%
$\geq 15,00$	13%

- De très belles prestations.
- Des exposés construits, fondés sur le texte fourni.
- Discussion autour de dimensions éthiques, sociétales, environnementales, économiques ou juridiques bien traitée.
- Faiblesses sur certains algorithmes du programme du lycée.
- L'illustration informatique doit être pertinente (attentes en programmation modestes).

Travaux pratiques 1/2

L'épreuve vise à s'assurer de la capacité du candidat à analyser et planifier son travail, à développer en conditions quasi-réelles, avec des outils standards, un code de qualité, à analyser un code produit par d'autres et à exposer et justifier les choix effectués en développant en particulier des problématiques de conception, de qualité du code et de garanties de comportement correct.

Le candidat reçoit un sujet (exemples sur <https://agreg-info.org/>).

Structure de l'oral : 30' de présentation (dont la présentation du code et l'**audit de code**); 30 minutes de questions.

Le jury attend

- du code pédagogique,
- une démonstration du fonctionnement.

Travaux pratiques 2/2

- Présents : 54
- Meilleure note : 20
- Moyenne : 10,85, écart-type : 4,73

$\geq 5,00$	87%
$\geq 7,60$	75%
$\geq 10,00$	57%
$\geq 10,33$	50%
$\geq 14,50$	25%
$\geq 15,00$	13%

- De très belles prestations.
- Production de code clair et pédagogique.
- Ce n'est pas un concours de vitesse de développement.
- Certains ont survolé les erreurs du code proposé (souvent de façon très partielle), puis proposé leur propre version du code demandé, alors que le but est de comprendre et d'aider à corriger les erreurs d'un code écrit par un élève.

Phase de questions