



ACADEMIE
DE TOULOUSE

*Liberté
Égalité
Fraternité*

WEBINAIRE

Nouveau format - Partie Pratique Epreuve NSI – session 2026

Mercredi 26 novembre

mercredi 26/11/2025

après-midi

5362



[Lien stagiaires](#)



[Lien intervenants](#)



Un nouveau texte

BO n°31 du 21 aout 2025 (note service 4/07/25)

OBJECTIFS

L'épreuve porte sur le programme de l'enseignement de spécialité de la classe de terminale en vigueur. Les notions du programme de la classe de première en vigueur non approfondies en classe de terminale, doivent être connues et mobilisables. Elles ne peuvent cependant pas constituer un ressort essentiel du sujet.

NATURE DE L'ÉPREUVE

L'épreuve terminale obligatoire de spécialité est composée de deux parties : une partie écrite et une partie pratique, chacune notée sur 20. La note de la partie écrite a un coefficient de 0,75 et celle de la partie pratique a un coefficient de 0,25. La note globale de l'épreuve est donnée sur 20 points.

Nouveau format épreuve pratique

Souhait du ministère d'un format « commun » pour les ECE

2. Partie pratique

Durée : 1 heure

Modalités

La partie pratique consiste à programmer sur ordinateur une application informatique à partir d'un document fourni au candidat.

L'épreuve a pour objectif d'évaluer le niveau de maîtrise des compétences pratiques du candidat.

Cette partie est notée sur 20 points.

Le candidat est évalué sur la base d'un dialogue avec un professeur-examinateur. Un examinateur évalue au maximum quatre élèves simultanément. L'examinateur ne peut pas évaluer un élève qu'il a eu en classe durant l'année en cours.

Pour rappel :

Ancien format : 2 exercices sur 10 points

Exercice 1 : programmation d'un algorithme explicitement au programme

Exercice 2 : modifier/ compléter un programme qui est fourni au candidat

Nouveau format épreuve pratique

Exemple ECE - SVT

L'épreuve d'ECE a évolué avec le souci d'une continuité dans la formation de l'élève qui s'organise autour de cinq axes :

1. la primauté d'une activité pratique pour résoudre une question scientifique donnée. Cette activité pratique intègre ou non le numérique ;
2. le développement de l'autonomie des élèves tant conceptuelle (conception d'une stratégie ; choix du mode de communication), organisationnelle (gestion du temps/appels à l'examinateur) que manipulatoires ;
3. l'introduction de l'argumentation orale ;
4. le développement progressif d'une évaluation par compétences, la mise en place de curseurs, de niveau de maîtrise et une évaluation différenciée par l'introduction des aides mineures et majeures ;
5. la mise en place de l'éducation à la sécurité.

Partie A : appropriation du contexte ; stratégie initiale et activité pratique

Partie B : communication des résultats ; poursuite de la stratégie, ressources complémentaires et conclusion

Organisation de l'épreuve pratique

- **Publication en décembre 2025 d'une note de cadrage** concernant les modalités d'organisation
- Lien vers la banque des sujets accessible fin mars – début avril 2026 (soixantaine de sujets)
- Passation de l'épreuve du **mardi 2 au lundi 8 juin** – A confirmer en fonction de la note de cadrage
- **4 élèves au maximum par examinateur**

Objectifs de l'épreuve pratique

- L'épreuve pratique permet une **double** rétroaction que ne permet pas l'écrit :
 - la **rétroaction avec la machine**, par l'élaboration d'une solution en s'appuyant sur la stratégie « essai-erreur »;
 - la **rétroaction avec l'examinateur** qui permet d'adapter la difficulté des questions, et d'évaluer la bonne compréhension au-delà de la simple exécution du code.
- La présence de fichiers sur machine permet :
 - de **donner réellement du sens** au contexte et d'étudier des problèmes plus ambitieux ;
 - d'évaluer la capacité à lire, **comprendre du code existant** et à l'utiliser, le modifier ou l'étendre.

Structuration des sujets

Les sujets sont structurés en deux temps.

- **Premier temps** : production d'une ou plusieurs **fonctions simples** répondant à un cahier des charges précis et dont l'objectif est de s'assurer de la capacité des candidats à écrire en **autonomie** des programmes, tout en leur permettant de s'aider des fichiers existants au besoin.
- **Deuxième temps** : **exploitation et travail avec du code existant** pour des objectifs variés comme :
 - S'assurer du bon fonctionnement en produisant un **jeu de tests** complet (sujet 0-1);
 - **Étendre le comportement** du programme pour des cas de figure non prévus ;
 - Observer et **corriger une erreur** du programme :
 - Elle peut être directement observable comme dans le sujet 0 – 1 (format non valide)
 - Logique comme dans le sujet 0 – 3 où le codage n'est pas toujours réversible
 - Plus subtile comme dans le cas du biais génré du sujet 0 – 2 ;
- Les deux parties partagent un cadre mais doivent pouvoir être **traitées indépendamment**.

Le contexte du sujet – Consignes à destination des concepteurs

- Pour permettre de donner du sens à ces deux parties du sujet, il est nécessaire de partir d'un **contexte lié à un problème pertinent** pour en déduire ensuite le bon découpage...
 - ...plutôt que de partir de questions types et d'y superposer un contexte très factice.
- Un bon contexte d'exercice :
 - doit s'adresser à tous **et à toutes** ;
 - doit montrer que l'informatique a une réelle **utilité pour la science ou la société** ;
 - ne doit **pas créer de difficultés de compréhension** et il doit donc s'exprimer rapidement et ne pas induire de difficultés pour les élèves ;
 - intéresse et permet de justifier l'introduction de notions informatiques.
- **En résumé, PAS DE CONTEXTE FICTIF !**

Des points de vigilance pour les concepteurs

- La **description du cadre** doit être la plus synthétique possible afin que tous les élèves puissent rapidement s'engager dans le problème.
- Il est indispensable de partir d'un **code qui fonctionne** qui respecte les standards de qualité et qui évite les excès de Python, même dans le code non destiné aux élèves.
- **Concernant les bibliothèques supplémentaires**, il ne sera pas possible de les déployer partout : risque d'écartement du sujet pour ces raisons. Privilégier l'absence de dépendances quitte à proposer des modules minimalistes pour s'y substituer.

L'accompagnement des élèves durant l'épreuve

- Ce changement dans les sujets demande de repenser l'accompagnement des élèves.
- Il n'est attendu une pleine autonomie que des meilleurs élèves.
- **Les élèves ont besoin d'une assistance et d'un retour de l'examinateur.**
- Des interactions forcées ont lieu à différents moments sous forme d'**appels professeur**. Ils permettent :
 - de s'assurer que la question et le contexte a bien été compris ;
 - de s'assurer de la compréhension des propositions de solutions et de la capacité des élèves à les présenter oralement ;
 - d'adapter la question aux capacités des élèves :
 - Pour une question d'écriture de code (1^{ère} partie) on peut ainsi fournir progressivement une explication globale, un début de fonction ou même un squelette complet
 - Pour une question de la 2^e partie, le cheminement de l'élève est plus important que la réponse. L'échange fait partie de la consigne.

Evaluation par compétences

- **Quatre compétences du programme** sont évaluées directement par cette épreuve
 - **P** Concevoir des solutions algorithmiques et traduire un algorithme dans un langage de programmation.
 - **A** Faire preuve d'autonomie, d'initiative et de créativité.
 - **C** Comprendre et réutiliser des codes sources existants, développer des processus de mise au point et de validation de programmes.
 - **O** Présenter un problème ou sa solution, développer une argumentation dans le cadre d'un débat.
- **Toutes ces compétences doivent être évaluées.**
- **L'évaluation de ces compétences doit guider la conception des sujets.**



➤ Session 2025

Evaluation par compétences

Chaque exercice est noté sur 10 et doit faire l'objet d'une notation particulière.

L'aspect oral fait pleinement partie de l'épreuve pratique. Il est donc conseillé de poser une ou plusieurs questions sur le code pour chaque exercice.

Pour la programmation, pour chaque exercice, il faut donner à minima la moitié des points si le code est cohérent même si le programme ne « tourne » pas.

La grille ci-dessous est un exemple de grille de compétence pour aider les évaluateurs. Elle n'a aucun caractère prescriptif.

Critères d'évaluation	Définition du critère	Très insuffisant	Insuffisant	Satisfaisant	Très satisfaisant
Connaissance des savoir-faire techniques	Connaissance des concepts de base	Besoin permanent d'assistance	A besoin de consignes complémentaires et d'assistance ponctuelle	A rarement besoin de consignes complémentaires	Travaille de façon autonome
Qualité de mise en œuvre	Niveau de conformité des opérations réalisées	Fait fréquemment des erreurs, exige une surveillance permanente	Produit un travail qu'il faut contrôler régulièrement	Fait des erreurs minimes qu'il ou elle parvient à verbaliser et propose des solutions	Travaille sans erreur
Qualité du dialogue	Justification	Pas de réponse	Pas clair	Relativement clair mais manque parfois de précision	Démontre une capacité à reformuler pour bien se faire comprendre.

Evaluation par compétences – Exemple ECE SVT

Évaluation de la réalisation pratique

Elle est notée sur 9 points à l'aide d'un curseur à quatre niveaux qui intègre le nombre et la nature des aides apportées au candidat.

Description	Niveau	Points
Seul ou avec une aide mineure, le candidat obtient des résultats exploitables	A	9
Avec plus d'une aide mineure, il obtient des résultats exploitables	B	6
Avec une aide majeure, il obtient des résultats exploitables	C	3
Malgré toutes les aides apportées, il n'obtient pas de résultats exploitables. <i>Un document de secours est indispensable.</i>	D	0

Distinction entre aide majeure et mineure

- **Aide mineure** : le correcteur cherche, par un conseil ou une remarque, à améliorer la pratique du candidat. La démarche engagée est pertinente, mais nécessite d'être poursuivie. Le correcteur valide et renforce ainsi l'action entreprise. Par exemple, le rangement du poste de travail constitue une aide mineure.
- **Aide majeure** : la démarche du candidat est inopérante ou comporte une lacune de sécurité importante. Pour obtenir un résultat exploitable, le correcteur doit alors se substituer au candidat : réaliser le geste technique, fournir un document de secours ou intervenir pour garantir

Sujets 0

➤ Rappel : Mis en ligne sur le site EDUSCOL depuis le mercredi 8 octobre

Chères et chers collègues,

Les trois premiers sujets de la banque pour l'épreuve pratique du Bac NSI 2026 sont publiés à l'adresse

<https://eduscol.education.fr/2068/programmes-et-ressources-en-numerique-et-sciences-informatiques-voie-g>

Description du dossier

Le dossier fourni au candidat sur l'ordinateur comporte les éléments suivants :

- une version PDF de l'énoncé ;
- un code source de départ `cycle_menstruel.py`.

Préparation de l'environnement

Pour faire fonctionner le code fourni dans le dossier, les bibliothèques suivantes doivent être présentes : `ics`.



Compétences :

P

A

C

O

SUJET 1 – 1^{er} temps

Une année est dite bissextile lorsqu'elle comporte 366 jours au lieu de 365. Une année est bissextile si elle est divisible par 4 à l'exception des années divisibles par 100 sauf si ce sont des multiples de 400.

Par exemple :

- 2024 est une année bissextile car elle est divisible par 4 et pas par 100 ;
 - 2100 n'est pas une année bissextile car elle est divisible par 4 et par 100 mais pas par 400 ;
 - 2000 est une année bissextile car elle est divisible par 4, par 100 et par 400.
1. Écrire en Python une fonction nommée `est_bissextile` qui prend en paramètre un entier correspondant à une année et qui renvoie un booléen indiquant si elle est bissextile, en appliquant la règle donnée ci-dessus.

Appel 1

Appeler le professeur en cas de difficulté de compréhension du codage.

2. Écrire en Python une fonction nommée `determiner_phase` qui prend en paramètre un entier (compris entre 1 et 28 inclus) qui correspond au jour d'un cycle et qui renvoie un entier correspondant au numéro de la phase associée. À l'aide d'une assertion, on garantira que l'entier donné en argument est compris entre 1 et 28 inclus.

Appel 2

Appeler le professeur pour lui présenter votre fonction et son fonctionnement ou en cas de difficultés.



SUJET 1 – 2^{ème} Temps

3. La fonction `ajouter_jours`, dont le code est déjà fourni, prend en paramètres une date et un entier représentant un nombre de jours. Elle renvoie la nouvelle date obtenue après ajout de ces jours.

Compléter la fonction `test_ajouter_jours` en ajoutant au moins trois autres tests pertinents. Pour chaque test ajouté, une brève justification doit être donnée afin d'expliquer pourquoi ce cas est important à vérifier.

On complète les nombres strictement inférieurs à 10 par un 0 pour s'assurer que la valeur de `DTSTART` soit toujours de longueur 8. Ainsi, la date du 3 juillet 2026 s'écrit sous la forme `DTSTART : 20260703`.

4. La fonction `calendrier_cycles`, présente dans le fichier fourni, prend en paramètre une date correspondant au premier jour des dernières règles et renvoie la liste chronologique des dates de début de règles qui se présentent dans les 100 jours suivant cette date, date incluse, au format `iCalendar` sous la forme d'une chaîne de caractères.

Observer avec la fonction `test_calendrier_cycles` que le calendrier renvoyé par la fonction `calendrier_cycles` n'est pas dans un format valide.

Identifier le problème dans la fonction `calendrier_cycles`, proposer une démarche de résolution et la mettre en œuvre.

Appel 3

Appeler le professeur pour lui présenter votre démarche ou en cas de difficultés.



ACADEMIE DE TOULOUSE

Liberté
Égalité
Fraternité

Compétences :

- **P** l'élève est capable de mettre en œuvre le décodage et de le programmer
- **A** l'élève est capable de lire le document, de programmer seul et de tester ses fonctions
- **C** l'élève est capable de s'inspirer du codage en observant le fichier, compléter la banque de tests ...
- **O** qu'il soit bloqué ou non, l'élève est capable d'expliquer sa situation, il prend en compte les suggestions du professeur

SUJET 3 – 1^{er} Temps

1. Déterminer si la liste obtenue par codage RLE est forcément de longueur inférieure ou égale à la liste de départ.

Appel 1

Appeler le professeur en cas de difficulté de compréhension du codage.

2. En étudiant bien la fonction `codage_rle` qui réalise le codage, écrire le corps de la fonction `decodage_rle` qui réalise le décodage d'une liste. Des tests sont fournis dans la fonction `test_codage()`, on pourra les compléter.

Appel 2

Appeler le professeur pour lui présenter votre fonction et son fonctionnement ou en cas de difficultés.



- C l'élève est capable d'identifier le problème et d'aller lire dans le code fourni l'endroit où il peut se produire
- A l'élève est capable de faire des tests, d'observer seul les problèmes, de procéder à des modifications
- O qu'il soit bloqué ou non, l'élève est capable d'expliquer sa situation, il prend en compte les suggestions du professeur

SUJET 3 – 2^{ème} Temps

3. Pour tester le codage sur une image, on peut utiliser la fonction fournie `encoder_decoder_image` qui effectue le codage puis le décodage d'une image pour l'enregistrer à nouveau dans un fichier. Utiliser cette fonction sur les images `bac_nsi_32.png` et `bac_nsi_256.png` et observer la différence de comportement.
4. Le problème précédent est lié au fait que sur des grandes images, il est possible d'avoir plus de 255 pixels de la même couleur. Proposer une démarche de résolution de ce problème qui modifie les fonctions d'encodage et de décodage, puis l'implémenter.

Appel 3	Appeler le professeur pour lui présenter votre démarche ou en cas de difficultés.
----------------	---



SUJET 2 – 1^{er} temps

1. Écrire le code de la fonction `salaire_moyen_condition` qui prend en paramètres un tableau d'employés dans le format décrit ci-dessus, le nom d'un des trois champs 'experience', 'etudes' ou 'sexe' ainsi qu'une valeur et qui renvoie un nombre flottant (type `float`) correspondant au salaire moyen des employés dont la valeur associée au champ est la valeur fournie.
2. Écrire en Python une fonction nommée `effectif_par_sexe` qui prend en paramètre un tableau non vide d'employés et qui renvoie un dictionnaire ayant deux clés 'F' et 'H' associées respectivement à l'effectif des femmes et à l'effectif des hommes employés.

Par exemple, avec le tableau `employes` précédent :

```
>>> effectif_par_sexe(employes)
{'F' :3, 'M' :3}
```

Appel 2

Appeler le professeur pour lui présenter votre fonction et son fonctionnement ou en cas de difficultés.



ACADEMIE DE TOULOUSE

Liberté
Égalité
Fraternité

SUJET 2 – 2ème temps

On définit l'écart de salaire moyen en pourcentage comme :

$$\text{écart} = \frac{\text{salaire moyen hommes} - \text{salaire moyen femmes}}{\text{salaire moyen hommes}} \times 100$$

Une fonction `calcule_ecart_sexe` est écrite dans le fichier `analyse.py`

Proposer une version corrigée de la fonction `ecart_salaire` qui valide ces tests et renvoie le bon écart. En déduire l'écart de salaire moyen dans les données complètes.

Appel 3

Appeler le professeur pour lui présenter vos tests et la correction proposée.

4. Afin de proposer un salaire d'embauche à un nouvel employé, le service informatique de l'entreprise a proposé d'utiliser l'algorithme des k-plus proches voisins et renvoyer la moyenne des salaires des trois employés aux caractéristiques les plus proches.

La fonction `salaire_par_proximité` effectue ce calcul pour faire une proposition.

Tester et comparer les salaires proposés aux deux futurs employés suivants :

```
{'experience': 3, 'etudes': 3, 'sexe': 'F'}  
{'experience': 3, 'etudes': 3, 'sexe': 'M'}
```

Identifier la source des écarts entre les deux propositions de salaire dans le programme et la corriger.

Appel 4

Appeler le professeur pour lui présenter vos tests, votre analyse des écarts et la correction proposée.

}. Expliquer pourquoi le code de cette fonction est incorrect et proposer quelques tests simples sous forme d'assertions qui permettent de mettre ce ou ces problèmes en évidence :

- vérifier que le résultat est `None` dans le cas où un seul sexe est présent ;
- vérifier qu'un écart de salaires exprimé en pourcentage soit toujours compris entre 0 et 100.



La semaine du numérique

Du 8 au 13 décembre 2025

Du 8 au 13 décembre 2025, les élèves de collège et lycée, leurs professeurs et leurs parents pourront bénéficier d'événements gratuits pour rencontrer des professionnels en classe, participer à des ateliers, à des visites d'entreprises, de centres de recherche et d'établissements de formation, assister à des webinaires ou des conférences....

Le programme est construit collectivement avec l'ensemble de l'écosystème numérique et éducatif.

Tous les événements doivent impérativement être référencés sur le site dédié : semaine-nsi.fr.

Dans le cadre de cette semaine, nous souhaitons réaliser, en lien avec la DRANE, un parcours ELEA à destination des élèves afin de présenter l'EDS NSI (enjeux, métiers, parcours ...)



ACADEMIE
DE TOULOUSE

Liberté
Égalité
Fraternité

La semaine du numérique

Du 8 au 13 décembre 2025

Occitanie

[Forum orientation NSI au lycée FOCH](#) - Rodez (12)

[Exposition interactive "Une histoire de code"](#) - Alès (30)

[Journée d'immersion : découverte du numérique](#) - Toulouse (31)

[La Mêlée Junior au Conseil Départemental de la Haute-Garonne](#) - Toulouse (31)

[Coding Club – Atelier d'initiation au code](#) - Montpellier (34)

Répondre Répondre à tous Transférer



GINESTE OLIVIER

[liste des lycées publics](#); [liste des lycées privés](#); [liste des collèges privés](#); [liste des collèges publics](#) ▾

03/11

TR: Semaine NSI/Webinaire: S'orienter vers le numérique - 10 déc. à 9H ou 14H/11 déc. à 18H - Inscription

Vous avez transféré ce message le 07/11/2025 10:33.
Ce message a été envoyé avec l'importance Haute.

Mesdames, Messieurs les chefs d'établissement,

Je vous sollicite afin de diffuser **très largement** auprès des **collégiens/lycéens/familles/enseignants/professeurs principaux** de votre établissement cette information concernant ces séances de webinaire « s'orienter vers le numérique : parcours et métiers » qui auront lieu pendant la semaine [du numérique et des sciences informatiques](#). (8 au 13 décembre 2025)

En vous remerciant pour votre collaboration,



Divers concours ...

- [Trophées NSI](#) (3/11/25 au 09/03/25)
- [Concours des Olympiades NSI en première](#)
(circulaire envoyée le 16/10 – date limite inscription : 6/03 – épreuve : 8/04)
- [La nuit du code](#)



**ACADEMIE
DE TOULOUSE**

*Liberté
Égalité
Fraternité*

TEMPS d'échanges

mercredi 26/11/2025

après-midi

5362



[Lien stagiaires](#)



[Lien intervenants](#)

