



SESSION 2025

CAPES ET CAFEP
Concours externe

Section

NUMERIQUE ET SCIENCES INFORMATIQUES

Épreuve disciplinaire appliquée

L'épreuve s'appuie sur un ou plusieurs documents pour un ou plusieurs niveaux de classe déterminés par le jury. Elle consiste en leur exploitation et leur analyse guidées par un questionnement précis. Elle vise à évaluer l'aptitude du candidat à mobiliser des savoirs disciplinaires et didactiques dans une séquence d'enseignement, ainsi que les capacités d'analyse, de synthèse et d'argumentation. Cette épreuve comprend une réflexion sur les dimensions éthiques, juridiques, économiques ou environnementales.

Le sujet est constitué d'un dossier pouvant comprendre un ou plusieurs énoncés d'exercices, des productions d'élèves, des documents institutionnels (extraits de programmes ou de ressources d'accompagnement), des extraits de manuels scolaires ou d'autres supports.

Il est demandé au candidat de concevoir une séquence portant sur un thème en lien avec les exercices du dossier. Lorsque le thème abordé se prête à une déclinaison progressive sur plusieurs niveaux d'enseignement, la cohérence didactique entre ces niveaux est valorisée.

Outre la résolution des exercices telle qu'elle pourrait figurer dans des cahiers d'élèves, il est attendu du candidat une évaluation des productions d'élèves, par exemple par l'identification et le traitement d'erreurs ou la valorisation de réussites, ou des propositions de remédiation ou d'approfondissement.

Durée : 5 heures

L'usage de tout ouvrage de référence, de tout dictionnaire et de tout matériel électronique est rigoureusement interdit.

Il appartient au candidat de vérifier qu'il a reçu un sujet complet et correspondant à l'épreuve à laquelle il se présente.

Si vous repérez ce qui vous semble être une erreur d'énoncé, vous devez le signaler très lisiblement sur votre copie, en proposer la correction et poursuivre l'épreuve en conséquence. De même, si cela vous conduit à formuler une ou plusieurs hypothèses, vous devez la (ou les) mentionner explicitement.

NB : Conformément au principe d'anonymat, votre copie ne doit comporter aucun signe distinctif, tel que nom, signature, origine, etc. Si le travail qui vous est demandé consiste notamment en la rédaction d'un projet ou d'une note, vous devrez impérativement vous abstenir de la signer ou de l'identifier. Le fait de rendre une copie blanche est éliminatoire.

Tournez la page S.V.P.

INFORMATION AUX CANDIDATS

Vous trouverez ci-après les codes nécessaires vous permettant de compléter les rubriques figurant en en-tête de votre copie. Ces codes doivent être reportés sur chacune des copies que vous remettrez.

CAPES EXTERNE NUMÉRIQUE ET SCIENCES INFORMATIQUES

► Concours externe du CAPES de l'enseignement public :

Concours	Section/option	Epreuve	Matière
E3E	6200E	102	9312

► Concours externe du CAFEP/CAPES de l'enseignement privé :

Concours	Section/option	Epreuve	Matière
E3F	6200E	102	9312

CAPES externe et CAFEP-CAPES
Section Numérique et Sciences Informatiques
Epreuve Disciplinaire appliquée – Session 2025

Préambule : cette épreuve est constituée de deux parties A et B indépendantes. Les réponses aux questions doivent être précises et rédigées avec soin. Certaines questions demandent des productions d'enseignement à destination des élèves (cours, exercices, projets, activités sur machines, activités débranchées, etc.). Ces questions devront être traitées avec la plus grande attention.

Partie A : la photographie numérique

Le but de cette partie est d'étudier le traitement de la photographie numérique à travers le continuum des programmes de SNT, première NSI et terminale NSI.

Elle est composée de cinq parties indépendantes qui couvrent plusieurs notions des référentiels.

Partie 1 : L'éthique et la photographie numérique

1. Pour traiter les aspects éthiques à considérer lors de la prise et de la publication de photographies numériques et notamment pour les sensibiliser au harcèlement, vous décidez d'organiser une discussion avec la classe de SNT mais souhaitez leur faire faire des recherches sur la protection de la vie privée et la diffusion d'images sensibles.

Proposez un questionnaire (de cinq ou six questions) auquel devront répondre vos élèves en effectuant des recherches sur le Web autour des notions de vie privée, de publication, de photographie numérique, de droit à l'image...

2. Expliquez en quelques lignes en quoi la photographie numérique soulève des enjeux éthiques, et pourquoi il est important de sensibiliser les élèves à ces enjeux.

Partie 2 : Notions juridiques et la cyberviolence

Nous nous intéressons ici à deux parties du cours de SNT : le Web et Les réseaux sociaux dont voici les deux extraits :

Contenus	Capacités attendues
Notions juridiques	Connaître certaines notions juridiques (licence, droit d'auteur, droit d'usage, valeur d'un bien).

Contenus	Capacités attendues
Cyberviolence	Connaître les dispositions de l'article 222-33-2-2 du code pénal. Connaître les différentes formes de cyberviolence (harcèlement, discrimination, sexting...) et les ressources disponibles pour lutter contre la cyberviolence

- Vous voudriez faire une activité sur les implications du droit d'auteur dans le domaine de la photographie numérique. Après avoir fait faire une recherche par petits groupes sur un cas réel de violation du droit d'auteur et les conséquences juridiques, vous aimeriez conclure l'activité par une synthèse sur ces deux points. Que proposeriez-vous ? L'article L335-2 sur les droits d'auteur se trouve en annexe.
- Vous organisez un débat sur la cyberviolence et le cyberharcèlement. Rédigez six questions pour orienter par une recherche la préparation du débat autour des capacités attendues du référentiel. L'article 222-33-2-2 du code pénal se trouve en annexe.

Partie 3 : La représentation matricielle d'une image et son traitement en itératif

Vous souhaitez faire travailler les élèves de SNT sur deux parties du référentiel : Notions transversales de programmation et La photographie numérique dont voici les deux extraits :

Contenus	Capacités attendues
Affectations, variables, Séquences, Instructions conditionnelles, Boucles bornées et non bornées, Définitions et appels de fonctions	Écrire et développer des programmes pour répondre à des problèmes et modéliser des phénomènes physiques, économiques et sociaux.

Contenus	Capacités attendues
Traitement d'image	Traiter par programme une image pour la transformer en agissant sur les trois composantes de ses pixels.

Et vous souhaitez faire travailler les élèves de première NSI sur les types construits et notamment sur cet extrait de programme :

Contenus	Capacités attendues
Tableau indexé, tableau donné en compréhension	Utiliser des tableaux de tableaux pour représenter des matrices : notation $a[i][j]$.

5. Comment peut-on représenter une image en niveaux de gris à l'aide de matrices en Python ? Quelle serait la notation pour accéder à un pixel spécifique de l'image ?
6. Comment, dans la représentation matricielle classique d'une image, le codage d'une image en couleur diffère-t-il de celui d'une image en niveaux de gris ?
7. Vous souhaitez introduire le cours sur le traitement d'images en seconde SNT. Proposez une activité débranchée pour leur faire découvrir la notion de négatif et la notion de niveau de gris pour une image. On pourra s'aider si besoin d'un quadrillage représentant une image.
8. Créez une fiche méthode pour l'utilisation du module Image de la bibliothèque PIL (pillow) à destination de la classe de seconde SNT. Un extrait de la bibliothèque pillow est fourni en annexe.
9. On vous propose un code dont vous vous inspirez pour le proposer à la classe de seconde SNT.

Proposez des questions pour guider vos élèves dans leur travail pour compléter les trous de ce code.

Puis détaillez une correction de vos questions, ainsi que le code correctement complété et commenté, tels que vous les donneriez à des élèves de seconde.

Code d'obtention d'un négatif :

```
from PIL import Image

# Ouvre l'image
img = Image.open(_____)

# Parcours tous les pixels de l'image
for y in range(img.height):
    for x in range(img.width):
        # Obtenir la couleur du pixel (R, G, B)
        r, g, b = img.getpixel((x, y))

        # Inverser les valeurs des couleurs pour obtenir un négatif
        r = _____
        g = _____
        b = _____

        # Modifier le pixel avec les nouvelles valeurs
        img.putpixel((x, y), (r, g, b))

# Sauvegarder l'image négative
img.save(_____)
```

10. Préparez les différentes étapes d'un cours aboutissant au programme précédent : vous détaillerez l'introduction à ce travail, les objectifs et le nombre de séances, ainsi que le travail sur la notion de double boucle.
11. Proposez un code pour obtenir une image en niveau de gris à partir d'une image. L'image de départ sera en RGB et sera notée : "image.jpg".
12. Afin d'introduire les tableaux de tableaux en première NSI, vous souhaitez mettre en place une auto-évaluation sous forme de QCM. Proposez 3 questions / réponses qui vous indiqueront la compréhension des élèves sur cette notion.
13. Un élève de première NSI vous rend une partie de son travail sur les tableaux de tableaux. Il vous indique que son script a un comportement qu'il juge étrange, car en modifiant un tableau de pixels (image2), l'autre tableau (image) est également modifié. Expliquez-lui la raison de ce comportement et proposez une correction.

```

image = [ [255,0,0],[0,255,0], [0,0,255],[255,255,255] ]
image2 = image

for x in range(len(image2)):
    for y in range(len(image2[x])):
        if image2[x][y] == 255:
            image2[x][y] = 125

```

14. Vous décidez, ensuite, de produire une petite activité sur les carrés magiques. Vous voulez vous inspirer d'un exercice du site France-IOI qui vous propose :

Entrée

La première ligne de l'entrée contient un entier N : le nombre de cases du côté de la grille de nombres.

Chacune des N lignes suivantes contient N entiers séparés par des espaces : les nombres d'une ligne de la grille.

Sortie

Vous devez afficher une ligne sur la sortie, contenant le mot "yes" si le carré fourni est un carré magique, et "no" sinon.

Définition d'un carré magique

Un **carré magique** est une grille ($N \times N$) dans laquelle :

- Tous les nombres sont uniques de 1 à N .
- La somme des nombres de chaque ligne, de chaque colonne et des deux diagonales est identique.

La somme commune à toutes ces lignes, colonnes et diagonales est appelée **constante magique**. Pour un carré magique de taille ($N \times N$), cette constante est égale à : $S = \frac{N(N^2+1)}{2}$

Exemple*entrée :*

3

6 1 8

7 5 3

2 9 4

Sortie :

yes

Décrivez le déroulement de l'activité qui permettra aux élèves de se familiariser avec les tableaux de tableaux. Pour cela, comment modéliser l'entrée ? Donner les différentes fonctions à créer pour décomposer le problème.

15. Vous aimeriez amener les élèves à créer un damier noir et blanc, de dimension 600 x 400, et dont chaque carré à un côté de longueur 20 pixels, à l'aide d'une fonction dont le début du code est le suivant :

```
from PIL import Image
def damier() :
    damier = Image.new('RGB', (600,400))
    for x in range(600) :
        for y in range(400) :
            ...
```

Expliquez comment amener les élèves à calculer le numéro de colonne et de ligne et à déterminer la couleur du pixel dans la fonction damier ci-dessus.

Partie 4 : La rotation des images

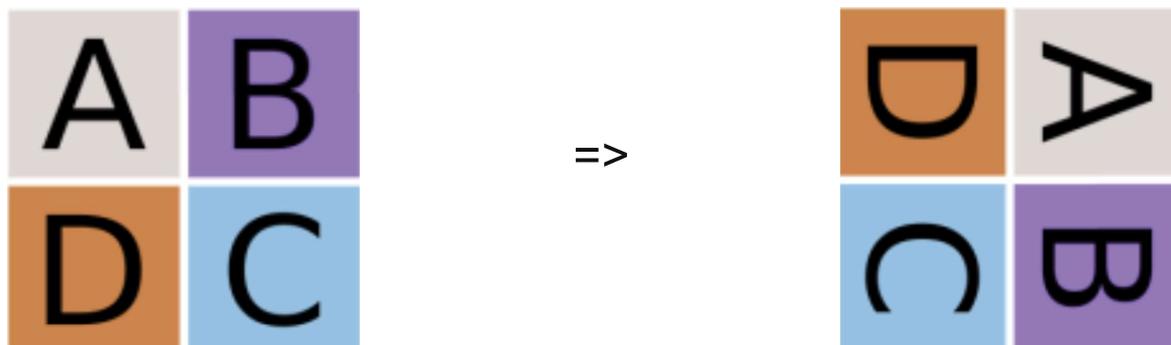
Nous nous intéresserons dans cette partie au programme de terminale et en particulier :

Contenus	Capacités attendues
Méthode « diviser pour régner ».	Écrire un algorithme utilisant la méthode « diviser pour régner ».

et

Contenus	Capacités attendues
Récurtivité.	Écrire un programme récursif. Analyser le fonctionnement d'un programme récursif.

16. Expliquez la méthode “diviser pour régner” dans le cadre de la rotation d’un quart de tour d’une image et en quoi cet algorithme a un coût en mémoire constant. On pourra utiliser l’image carrée suivante (celle de gauche) qui deviendrait celle de droite (après rotation d’un quart de tour) :



17. Vous souhaitez mettre en œuvre, sous forme de mini-projet, la rotation d’une image d’un quart de tour avec une “méthode diviser pour régner”. Programmez plusieurs fonctions pour implémenter cet algorithme en récursif. On se limitera au cas où l’image est de taille NxN où N est une puissance de 2.

On pourra s’aider des fonctions suivantes (les arguments des fonctions sont décrits sous l’encadré) :

```
from PIL import Image

# Cette fonction échange deux pixels dans l'image source.
def echangePixel(source, x1, y1, x2, y2):
    pass

# Cette fonction échange deux quadrants dans l'image source.
def echangeQuadrant(source, x1, y1, x2, y2, n):
    pass

# Cette fonction effectue une rotation d'un quart de tour d'un quadrant de l'image source.
def tourneQuadrant(source, x, y, n):
    pass

# Cette fonction effectue une rotation d'un quart de tour de l'image source.
def rotationDpR(source):
    largeur, hauteur = source.size # dimensions de l'image
    tourneQuadrant(source, 0, 0, largeur) # Effectue une rotation sur l'ensemble de l'image
```

avec

- `echangePixel(source, x1, y1, x2, y2)`
 - `source` : C’est l’image d’origine que nous manipulons pour échanger les pixels.

- x1 : Coordonnée en x du premier pixel à échanger (colonne). Cela représente la position horizontale (axe x) du premier pixel.
- y1 : Coordonnée en y du premier pixel à échanger (ligne). Cela représente la position verticale (axe y) du premier pixel.
- x2 : Coordonnée en x du deuxième pixel à échanger (colonne). Cela représente la position horizontale (axe x) du deuxième pixel.
- y2 : Coordonnée en y du deuxième pixel à échanger (ligne). Cela représente la position verticale (axe y) du deuxième pixel.
- echangeQuadrant(source, x1, y1, x2, y2, n)
 - source : L'image source dans laquelle les quadrants vont être échangés.
 - x1 : Coordonnée en x du coin supérieur gauche du premier quadrant à échanger. C'est la position horizontale du coin supérieur gauche du premier quadrant à échanger.
 - y1 : Coordonnée en y du coin supérieur gauche du premier quadrant à échanger. C'est la position verticale du coin supérieur gauche du premier quadrant.
 - x2 : Coordonnée en x du coin supérieur gauche du deuxième quadrant à échanger. Position horizontale du coin supérieur gauche du deuxième quadrant à échanger.
 - y2 : Coordonnée en y du coin supérieur gauche du deuxième quadrant à échanger. Position verticale du coin supérieur gauche du deuxième quadrant.
 - n : Taille du côté des quadrants échangés (le quadrant est de taille n x n). La taille de chaque quadrant à échanger (chaque quadrant est un carré de côté n).
- tourneQuadrant(source, x, y, n)
 - source : L'image source dans laquelle nous effectuons la rotation d'un quadrant.
 - x : Coordonnée en x du coin supérieur gauche du quadrant à faire tourner. Position horizontale du coin supérieur gauche du quadrant à faire tourner.
 - y : Coordonnée en y du coin supérieur gauche du quadrant à faire tourner. Position verticale du coin supérieur gauche du quadrant à faire tourner.
 - n : Taille du côté du quadrant à faire tourner (le quadrant est de taille n x n). La taille du quadrant à faire tourner, cette fonction va travailler sur des sous-parties de l'image.
- rotationDpR(source)
 - source : L'image complète sur laquelle la rotation d'un quart de tour va être appliquée.

18. Proposez un code plus simple, non récursif et de même complexité spatiale

Partie 5: Les filtres d'image en Programmation Orientée Objet (POO)

Nous nous intéresserons dans cette partie au programme de terminale et en particulier :

Contenus	Capacités attendues
Vocabulaire de la programmation objet : classes, attributs, méthodes, objets.	Écrire la définition d'une classe. Accéder aux attributs et méthodes d'une classe.

On cherche à faire implémenter quelques filtres dans le cadre du programme de terminale.

19. Proposez une activité d'introduction à la POO pour permettre aux élèves à s'appropriier ces notions : classes, attributs, méthodes, objets.
20. Vous remarquez que les élèves confondent les classes et les instances (objet). Comment pourriez-vous les aider à préciser ce vocabulaire ?
21. Comment leur expliquer le mot 'self' qui apparaît en POO ?

Vous décidez de proposer aux élèves une activité qui lie la POO et les filtres des images dont le diagramme de classe est le suivant :

<p>Classe Filtre:</p> <ul style="list-style-type: none">● constructeur :<ul style="list-style-type: none">○ fichier : string○ img : Image○ pix : list● Méthodes:<ul style="list-style-type: none">○ <code>__init__(fichier: string)</code>○ <code>taille()</code> : tuple○ <code>largeur()</code> : int○ <code>hauteur()</code> : int○ <code>nombre_pixels()</code> : int○ <code>get_pixel(col: int, ligne: int)</code> : tuple

- img permet l'ouverture de l'image
- pix permet d'accéder à un pixel de l'image que l'on a ouvert avec img
- `taille()` retourne la taille en pixels d'une image sous forme de tuple largeur, hauteur
- `largeur()` retourne la largeur d'une image en pixels
- `hauteur()` retourne la hauteur d'une image en pixels
- `nombre_pixels()` retourne le nombre total de pixels d'une image
- `get_pixel (col, ligne)` retourne la valeur du pixel de coordonnées (col,ligne), ou None si les coordonnées sont incorrectes.

22. Vous voulez proposer des aides supplémentaires et/ou du code pour accompagner les élèves dans cette activité, afin de les guider de manière progressive et claire.

Proposez des supports qui permettent aux élèves de :

- Comprendre précisément les objectifs et attendus : quelles méthodes doivent être complétées.

Comment utiliser chaque méthode (exemples attendus).

- S'assurer de la bonne compréhension des méthodes et attributs : compléter une structure de code avec des trous en respectant les étapes données. Utiliser des questions progressives qui guideront les élèves sur l'utilisation des attributs et méthodes de la classe.

23. Le but, à présent, est de faire faire une méthode `couleur_vers_gris(fichier: string)` qui permet de transformer la photo couleur en nuance de gris. Comment amener les élèves à faire une double boucle dans le codage de cette méthode ?

24. Proposez un algorithme en pseudo-code qui permette à l'élève d'implémenter la méthode : `couleur_vers_gris(fichier: string)` et décrivez ce qu'il fait.

25. Proposez une implémentation de l'algorithme précédent.

Partie B : optimisation par la programmation dynamique

Vous souhaitez aborder la notion de programmation dynamique avec comme objectifs :

- Comprendre le concept de programmation dynamique.
- Reconnaître les problèmes pouvant être résolus efficacement avec la programmation dynamique.
- Savoir appliquer la méthode de résolution de problèmes par programmation dynamique.

Vous avez décidé d'accompagner vos élèves de terminale à travers une activité concrète. Étant donné que votre classe présente une hétérogénéité de niveaux, cette activité sera abordée en plusieurs séances. Cela vous permettra d'adapter les séances en fonction des difficultés rencontrées par chaque élève.

La propriétaire d'un cheval se penche sur le calendrier des courses hippiques de la saison à venir :

- chaque jour il y a une course hippique à laquelle il peut participer ;
- chacune des courses hippiques offre un prix pour le vainqueur ;
- après avoir participé à une course, le cheval doit se reposer et ne peut participer à une nouvelle course que quatre jours plus tard.

La propriétaire cherche à choisir les courses auxquelles il participera en maximisant

le montant total des prix qu'il pourra gagner s'il est vainqueur.

On suppose que le calendrier des courses est implémenté par un tableau indexé 'calendrier' pour lequel la valeur d'indice 'i' est le prix offert pour la course du 'i'ème jour de la saison (en commençant à zéro).

Le planning des courses choisies est implémenté par un tableau indexé 'planning' pour lequel la valeur d'indice 'i' est le jour choisi de la saison (en commençant à zéro). Cette valeur doit correspondre à un jour référencé dans le calendrier.

Exemple :

calendrier = [2, 1, 10, 6, 8, 20, 22, 10, 4, 20, 2, 4, 14, 2, 6] est de taille 15.

La propriétaire peut choisir comme planning [0, 4, 8, 12] auquel cas le montant total des prix est $2 + 8 + 4 + 14 = 28$.

Partie 1 : Planning

26. Afin de faire comprendre la contrainte concernant uniquement le repos du cheval dans le planning choisi, proposez quatre exemples pertinents que les élèves doivent résoudre à la main.

Vous proposez aux élèves d'écrire une fonction 'planning_correct' qui admet comme argument 'planning' une liste représentant un planning et qui renvoie 'True' si le planning est correct sinon 'False'. Dans un premier temps et afin de simplifier l'écriture de cette fonction, vous proposez de ne considérer que les jours de repos comme contrainte.

27. Écrivez cette fonction.

28. Vous trouverez ci-dessous des réponses données par des élèves.

Elève 1

```
def planning_correct(planning):  
    for i in range(len(planning)-1):  
        p = planning[i+1] - planning[i]  
        return p > 4
```

```
planning_correct([0, 4, 8, 12])
```

False

Elève 2

```
def planning_correct(planning):  
    for i in planning:  
        if i+4 >= i:  
            return False  
        else:  
            return True
```

Quelles sont les erreurs commises par chaque élève ? Quels commentaires proposez-vous sur la copie afin de lui suggérer une piste de réflexion ?

29. Complétez la fonction 'planning_correct' de la question 27 qui admet comme arguments 'planning' une liste représentant un planning et 'calendrier' une liste représentant un calendrier afin de tenir compte de toutes les contraintes et qui renvoie 'True' si le planning est correct sinon 'False'.
30. Proposez quatre plannings avec le calendrier = [2, 1, 10, 6, 8, 20, 22, 10, 4, 20, 2, 4, 14, 2, 6] afin que les élèves vérifient leur fonction.
31. Au regard des erreurs des élèves et dans le cadre d'une remédiation, quelles sont les trois notions importantes que vous pourriez aborder avec les élèves ?
32. Proposez quatre exercices courts afin de vérifier les acquis des élèves.

Partie 2 : Gain possible

La partie 1 est corrigée. La fonction 'planning_correct' est donnée aux élèves.

33. Vous proposez aux élèves d'écrire une fonction 'gain_possible' qui admet comme arguments 'calendrier' une liste représentant un calendrier et 'planning' une liste représentant un planning et qui renvoie le gain possible dans le cas où le planning est correct sinon -1.

Vous trouverez ci-dessous des réponses données par des élèves.

Elève 1

```
def gain_possible(calendrier, planning):
    assert isinstance(calendrier,list) and isinstance(planning,list)
    if planning_correct == True :
        for g in range(calendrier):
            for i in range(planning):
                return calendrier[planning[i]]
    else :
        return -1
```

Elève 2

```
def gain_possible(calendrier, planning):
    gain = 0
    if planning_correct(planning) == True:
        for elem in calendrier:
            gain += elem
        return gain
    return -1
```

Elève 3

```
def gain_possible(calendrier, planning):
    for i in planning:
        for elem in calendrier:
            if
                gain = gain + elem
            else:
```

Indiquez les erreurs commises par chaque élève et proposez une correction en tenant compte ces erreurs, en justifiant vos choix.

Partie 3 : Résolution par un algorithme glouton

Nous nous intéresserons dans cette partie à un algorithme vu en classe de première

Contenus	Capacités attendues
Algorithmes gloutons	Résoudre un problème grâce à un algorithme glouton

Vous décidez de reprendre un exemple classique de rendu de monnaie vu en première afin de préparer les élèves à une solution gloutonne de l'activité.

On rappelle que le problème du rendu de monnaie consiste à trouver le nombre minimal de pièces ou de billets nécessaires pour payer une somme donnée.

On veut programmer une caisse automatique pour qu'elle rende la monnaie de façon optimale, c'est-à-dire avec le nombre minimal de pièces et de billets.

Les valeurs des pièces et billets à disposition sont : 1, 2, 5, 10, 20, 50, 100 et 200 euros. On suppose que la machine dispose d'autant d'exemplaires de pièces et billets que nécessaires pour rendre la monnaie.

34. Qu'est-ce qu'un algorithme glouton ?
35. Détaillez l'algorithme dans le cas du rendu de monnaie.
36. Un élève vous demande si la solution proposée est optimale pour tous les systèmes monétaires. Donnez un contre-exemple.
37. Comment expliquez-vous à vos élèves la différence fondamentale entre l'approche de l'algorithme glouton et l'approche par force brute pour résoudre le problème du rendu de monnaie ?
38. Détaillez l'algorithme glouton dans le cas du calendrier des courses hippiques.
39. Pour résoudre ce problème, un élève propose d'écrire une fonction qui mettra à jour le calendrier en mettant à 0 la valeur de l'indice 'jour' choisie ainsi que les 3 valeurs des indices avant et après si possible. Et pour cela, il propose cette fonction :

```
def supprime_jours(calendrier, jour):  
    # Test validation planning  
    assert 0 <= jour < len(calendrier), "jour non valide"  
    for i in range(0,4):  
        if jour + i < len(calendrier) and jour - i >= 0:  
            calendrier[jour + i] = 0  
            calendrier[jour - i] = 0  
    return calendrier
```

Quelle est l'erreur de l'élève ? Proposez un exemple de calendrier et jour pour lui faire comprendre son erreur en le justifiant.

40. Vous décidez de proposer une évaluation sous la forme d'une épreuve pratique de terminale (voir annexe).

L'exercice 1 portera sur la notion de listes.

L'exercice 2 s'appuiera sur le programme suivant.

```
def max_gain_courses(prix_courses):
    total_gain = 0
    while sum(prix_courses) != 0:
        meilleur_jour = prix_courses.index(max(prix_courses))
        total_gain += prix_courses[meilleur_jour]
        prix_courses = supprime_jours(prix_courses, meilleur_jour)
    return total_gain
```

Proposez deux exercices en tenant compte des attendus. Vous devrez justifier brièvement vos choix.

Partie 4 : Récursivité

On donne la fonction récursive `total_prix` qui admet comme arguments 'calendrier' une liste représentant un calendrier et `jour` un entier représentant l'indice de la dernière course possible et qui renvoie le gain maximum.

Ainsi `total_prix(calendrier, 9)` calcule le gain maximum possible lorsqu'on choisit les 10 premières dates du calendrier.

```
def total_prix(calendrier, jour):
    if jour < 4:
        prix_total_max = max(calendrier[i] for i in range(jour + 1))
        return prix_total_max
    else:
        prix_total_max = max(total_prix(calendrier, jour - 1),
                             total_prix(calendrier, jour - 4) + calendrier[jour])
    return prix_total_max
```

41. Intéressons-nous au test d'arrêt :

```
if jour < 4:
    prix_total_max = max(calendrier[i] for i in range(jour + 1))
    return prix_total_max
```

Pourriez-vous expliquer le choix de cette condition `jour < 4` et le calcul de la variable `prix_total_max` ?

Pourquoi cette condition est-elle nécessaire et comment influence-t-elle le comportement de la fonction ?

42. Dessinez soigneusement l'arbre des appels récursifs lorsque l'appel principal est `total_prix(calendrier, 9)`. Combien d'appels sont effectués en tout (appel principal compris) ?
43. Comment peut-on modifier la fonction `total_prix` afin de réduire le nombre d'appels ?

Partie 5 : Programmation dynamique

Nous nous intéresserons dans cette partie à la programmation dynamique de la classe de terminale :

Contenus	Capacités attendues
Programmation dynamique	Utiliser la programmation dynamique pour écrire un algorithme.

Il existe deux facettes équivalentes de la programmation dynamique :

- La programmation descendante récursive avec mémoïsation appelée top-down ;
- La programmation ascendante itérative, appelée bottom up.

44. Comment expliquez-vous aux élèves, en quelques phrases, ces deux approches. Vous illustrerez vos explications par des exemples.

45. Vous proposez aux élèves la fonction total répondant au paradigme de programmation dynamique ci-dessous et l'affichage des différents calculs.

```
cal = [2, 25, 30, 2, 15, 2, 25, 6, 15, 25, 6, 2, 6, 6, 6, 2, 15, 6, 6, 6, 15, 15, 25, 25, 6, 15]

def maxi_debut(cal):
    maxi = cal[0]
    m =[maxi]
    for i in range(1,4):
        maxi = max(maxi,cal[i])
        m.append(maxi)
    return m

def total(cal,n):
    t = maxi_debut(cal)
    for i in range(4,n + 1):
        t.append (max( t[i - 1] , cal[i] + t[i - 4]))
    print(t)
    return t

total(cal,25)
```

```

[2, 25, 30, 30, 30]
[2, 25, 30, 30, 30, 30]
[2, 25, 30, 30, 30, 30, 55]
[2, 25, 30, 30, 30, 30, 55, 55]
[2, 25, 30, 30, 30, 30, 55, 55, 55]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91, 91]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91, 91, 101]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91, 91, 101, 101]
[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91, 91, 101, 101, 106]

[2, 25, 30, 30, 30, 30, 55, 55, 55, 55, 61, 61, 61, 61, 67, 67, 76, 76, 76, 91, 91, 101, 101, 101, 106]

```

En quels termes expliquez-vous aux élèves, les valeurs obtenues à la première ligne de l'affichage ?

Choisissez des lignes pertinentes consécutives de l'affichage pour expliquer les principes de la programmation dynamique.

Annexes

Article 222-33-2-2

Modifié par LOI n°2022-299 du 2 mars 2022 - art. 13

Le fait de harceler une personne par des propos ou comportements répétés ayant pour objet ou pour effet une dégradation de ses conditions de vie se traduisant par une altération de sa santé physique ou mentale est puni d'un an d'emprisonnement et de 15 000 € d'amende lorsque ces faits ont causé une incapacité totale de travail inférieure ou égale à huit jours ou n'ont entraîné aucune incapacité de travail.

L'infraction est également constituée :

a) Lorsque ces propos ou comportements sont imposés à une même victime par plusieurs personnes, de manière concertée ou à l'instigation de l'une d'elles, alors même que chacune de ces personnes n'a pas agi de façon répétée ;

b) Lorsque ces propos ou comportements sont imposés à une même victime, successivement, par plusieurs personnes qui, même en l'absence de concertation, savent que ces propos ou comportements caractérisent une répétition.

Les faits mentionnés aux premier à quatrième alinéas sont punis de deux ans d'emprisonnement et de 30 000 € d'amende :

1° Lorsqu'ils ont causé une incapacité totale de travail supérieure à huit jours ;

2° Lorsqu'ils ont été commis sur un mineur ;

3° Lorsqu'ils ont été commis sur une personne dont la particulière vulnérabilité, due à son âge, à une maladie, à une infirmité, à une déficience physique ou psychique ou à un état de grossesse, est apparente ou connue de leur auteur ;

4° Lorsqu'ils ont été commis par l'utilisation d'un service de communication au public en ligne ou par le biais d'un support numérique ou électronique ;

5° Lorsqu'un mineur était présent et y a assisté.

Les faits mentionnés aux premier à quatrième alinéas sont punis de trois ans d'emprisonnement et de 45 000 € d'amende lorsqu'ils sont commis dans deux des circonstances mentionnées aux 1° à 5°.

Article L335-2

Modifié par LOI n°2016-731 du 3 juin 2016 - art. 44

Toute édition d'écrits, de composition musicale, de dessin, de peinture ou de toute autre production, imprimée ou gravée en entier ou en partie, au mépris des lois et règlements relatifs à la propriété des auteurs, est une contrefaçon et toute contrefaçon est un délit.

La contrefaçon en France d'ouvrages publiés en France ou à l'étranger est punie de trois ans d'emprisonnement et de 300 000 euros d'amende.

Seront punis des mêmes peines le débit, l'exportation, l'importation, le transbordement ou la détention aux fins précitées des ouvrages contrefaisants.

Lorsque les délits prévus par le présent article ont été commis en bande organisée, les peines sont portées à sept ans d'emprisonnement et à 750 000 euros d'amende.

Extrait de l'aide du module Image de la bibliothèque PIL

Help on module PIL.Image in PIL:

```
class Image(builtins.object)
```

```
| This class represents an image object. To create  
| :py:class:`~PIL.Image.Image` objects, use the appropriate factory  
| functions. There's hardly ever any reason to call the Image constructor  
| directly.
```

```
| * :py:func:`~PIL.Image.open`
```

```
| * :py:func:`~PIL.Image.new`
```

```
| Methods defined here:
```

```
| getpixel(self, xy)
```

```
| Returns the pixel value at a given position.
```

```
| :param xy: The coordinate, given as (x, y). See
```

```
| :ref:`coordinate-system`.
```

```
| :returns: The pixel value. If the image is a multi-layer image,  
| this method returns a tuple.
```

```
| load(self)
```

```
| Allocates storage for the image and loads the pixel data. In  
| normal cases, you don't need to call this method, since the  
| Image class automatically loads an opened image when it is  
| accessed for the first time.
```

```
| If the file associated with the image was opened by Pillow, then this  
| method will close it. The exception to this is if the image has  
| multiple frames, in which case the file will be left open for seek  
| operations. See :ref:`file-handling` for more information.
```

```
| :returns: An image access object.
```

```
| :rtype: :ref:`PixelAccess` or :py:class:`PIL.PyAccess`
```

```
| putpixel(self, xy, value)
```

```
| Modifies the pixel at the given position. The color is given as  
| a single numerical value for single-band images, and a tuple for  
| multi-band images. In addition to this, RGB and RGBA tuples are  
| accepted for P and PA images.
```

Note that this method is relatively slow. For more extensive changes, use `:py:meth:`~PIL.Image.Image.paste`` or the `:py:mod:`~PIL.ImageDraw`` module instead.

`:param xy`: The pixel coordinate, given as (x, y). See

`:ref:`coordinate-system``.

`:param value`: The pixel value.

`save(self, fp, format=None, **params)`

Saves this image under the given filename. If no format is specified, the format to use is determined from the filename extension, if possible.

Keyword options can be used to provide additional instructions to the writer. If a writer doesn't recognise an option, it is silently ignored. The available options are described in the `:doc:`image format documentation <../handbook/image-file-formats>`` for each writer.

You can use a file object instead of a filename. In this case, you must always specify the format. The file object must implement the ```seek```, ```tell```, and ```write``` methods, and be opened in binary mode.

`:param fp`: A filename (string), `pathlib.Path` object or file object.

`:param format`: Optional format override. If omitted, the format to use is determined from the filename extension.

If a file object was used instead of a filename, this parameter should always be used.

`:param params`: Extra parameters to the image writer.

`:returns`: None

`:exception ValueError`: If the output format could not be determined from the file name. Use the format option to solve this.

`:exception OSError`: If the file could not be written. The file may have been created, and may contain partial data.

`show(self, title=None)`

Displays this image. This method is mainly intended for debugging purposes.

This method calls `:py:func:`~PIL.ImageShow.show`` internally. You can use `:py:func:`~PIL.ImageShow.register`` to override its default behaviour.

The image is first saved to a temporary file. By default, it will be in PNG format.

`:param title`: Optional title to use for the image window, where possible.

Epreuve pratique de NSI

Partie pratique

Durée : 1 heure

Modalités

La partie pratique consiste en la résolution de deux exercices sur ordinateur, chacun étant noté sur 10 points.

Le candidat est évalué sur la base d'un dialogue avec un professeur-examineur.

Un examinateur évalue au maximum quatre élèves. L'examineur ne peut pas évaluer un élève qu'il a eu en classe durant l'année en cours.

L'évaluation de cette partie se déroule au cours du deuxième trimestre pendant la période de l'épreuve écrite de spécialité.

- Premier exercice

Le premier exercice consiste à programmer un algorithme figurant explicitement au programme, ne présentant pas de difficulté particulière, dont on fournit une spécification. Il s'agit donc de restituer un algorithme rencontré et travaillé à plusieurs reprises en cours de formation. Le sujet peut proposer un jeu de test avec les réponses attendues pour permettre au candidat de vérifier son travail.

- Deuxième exercice

Pour le second exercice, un programme est fourni au candidat. Cet exercice ne demande pas l'écriture complète d'un programme, mais permet de valider des compétences de programmation suivant des modalités variées : le candidat doit, par exemple, compléter un programme « à trous » afin de répondre à une spécification donnée, ou encore compléter un programme pour le documenter, ou encore compléter un programme en ajoutant des assertions, etc.