



# **Classes préparatoires aux grandes écoles**

## **Filière scientifique**

### **Voie Biologie, chimie, physique et sciences de la Terre (BCPST)**

#### **Annexe 4**

#### **Programmes d'informatique 1<sup>ère</sup> et 2<sup>nde</sup> années**

# Table des matières

<b>1</b>	<b>Programme du semestre 1</b>	<b>5</b>
<b>2</b>	<b>Programme du semestre 2</b>	<b>6</b>
2.1	Méthodes de programmation et dictionnaire . . . . .	6
2.2	Bases de données . . . . .	6
2.3	Graphes . . . . .	7
2.4	Méthodes numériques . . . . .	7
<b>3</b>	<b>Programme des semestres 3 et 4</b>	<b>8</b>
3.1	Méthodes numériques et statistiques . . . . .	8
3.2	Approfondissements des concepts informatiques . . . . .	9
<b>A</b>	<b>Langage Python</b>	<b>11</b>

## Introduction au programme

**Les objectifs du programme** Le programme d'informatique de BCPST s'inscrit entre deux continuités : en amont avec les programmes rénovés du lycée, en aval avec les enseignements dispensés dans les grandes écoles, et plus généralement les poursuites d'études universitaires. Il a pour objectif la formation de futurs ingénieures et ingénieurs, vétérinaires, enseignantes et enseignants, chercheuses et chercheurs et avant tout des personnes informées, capables de gouverner leur vie professionnelle et citoyenne en pleine connaissance et maîtrise des techniques et des enjeux de l'informatique et en la nourrissant par les habitudes de la démarche scientifique.

Le présent programme a pour ambition de poser les bases d'un enseignement cohérent et mesuré d'une science informatique encore jeune et dont les manifestations technologiques connaissent des cycles d'obsolescence rapide. On garde donc à l'esprit :

- de privilégier la présentation de concepts fondamentaux pérennes sans s'attacher outre mesure à la description de technologies, protocoles ou normes actuels;
- de donner aux futurs diplômées et diplômés les moyens de réussir dans un domaine en mutation rapide et dont les technologies qui en sont issues peuvent sauter brutalement d'un paradigme à un autre très différent;
- de préparer les étudiantes et étudiants à tout un panel de professions et de situations de la vie professionnelle qui les amène à remplir tour à tour une mission d'expertise, de création ou d'invention, de prescription de méthodes ou de techniques, de contrôle critique des choix opérés ou encore de décision en interaction avec des spécialistes;

**Compétences visées** Ce programme vise à développer les six grandes compétences suivantes :

**analyser et modéliser** un problème ou une situation, notamment en utilisant les objets conceptuels de l'informatique pertinents (table relationnelle, graphe, dictionnaire, etc.);

**imaginer et concevoir une solution**, décomposer en blocs, se ramener à des sous-problèmes simples et indépendants, adopter une stratégie appropriée, décrire une démarche, un algorithme ou une structure de données permettant de résoudre le problème;

**décrire et spécifier** les caractéristiques d'un processus, les données d'un problème, ou celles manipulées par un algorithme ou une fonction;

**mettre en œuvre une solution**, par la traduction d'un algorithme ou d'une structure de données dans un langage de programmation ou un langage de requête;

**justifier et critiquer une solution**, en développant des processus d'évaluation, de contrôle, de validation d'un code que l'on a produit;

**communiquer à l'écrit ou à l'oral**, présenter des travaux informatiques, une problématique et sa solution; défendre ses choix; documenter sa production et son implémentation.

La pratique régulière de la résolution de problèmes par une approche algorithmique et des activités de programmation qui en résultent constitue un aspect essentiel de l'apprentissage de l'informatique. Les exemples ou les exercices d'application peuvent être choisis au sein de l'informatique elle-même ou en lien avec d'autres champs disciplinaires.

**Sur les partis pris par le programme** Ce programme impose aussi souvent que possible des choix de vocabulaire ou de notation de certaines notions. Les choix opérés ne présument pas la supériorité de l'option retenue. Ils ont été précisés dans l'unique but d'aligner les pratiques d'une classe à une autre et d'éviter l'introduction de longues définitions récapitulatives préliminaires à un exercice ou un problème. De même, ce programme nomme aussi souvent que possible l'un des algorithmes possibles parmi les classiques qui répondent à un problème donné. Là encore, le programme ne défend pas la prééminence d'un algorithme ou d'une méthode par rapport à un autre mais il invite à faire bien plutôt que beaucoup.

**Sur les langages et la programmation** L'enseignement du présent programme repose sur un langage de manipulation de données (SQL) ainsi que le langage de programmation Python, pour lequel une annexe liste de façon limitative les éléments qui sont exigibles des étudiants. La poursuite de l'apprentissage du langage Python est vue en particulier par les étudiants pour adopter immédiatement une bonne discipline de programmation tout en se concentrant sur le noyau du langage plutôt que sur une interface de programmation applicative (API) pléthorique.

**Mode d'emploi** Ce programme a été rédigé par semestre pour assurer une certaine homogénéité de la formation. Le premier semestre permet d'asseoir les bases de programmation vues au lycée et les concepts associés. L'organisation de la progression au sein des semestres relève de la responsabilité pédagogique de la professeure ou du professeur et le tissage de liens entre les thèmes contribue à la valeur de son enseignement. Les notions étudiées lors d'un semestre précédent sont régulièrement revisitées tout au long des deux années d'enseignement. Le programme est présenté sous forme de tableaux à deux colonnes. Les premières colonnes - notions et thèmes - présentent les attendus exigibles du programme. Les secondes colonnes présentent des listes, sans aucun caractère impératif, d'exemples d'activités qui peuvent être proposées aux étudiants ainsi que des commentaires.

# 1 Programme du semestre 1

Les séances de travaux pratiques du premier semestre poursuivent les objectifs suivants :

- consolider l'apprentissage de la programmation en langage Python qui a été entrepris dans les classes du lycée;
- mettre en place un environnement de travail;
- mettre en place une discipline de programmation : spécification précise des fonctions et programmes, annotations et commentaires, jeux de tests;

La consolidation du langage Python portera principalement sur

- les variables
- les expressions et instructions
- les instructions conditionnelles
- les fonctions
- les instructions itératives
- la manipulation de quelques structures de données

Rappelons que l'annexe liste les éléments du langage Python qui sont exigibles des étudiants.

Le tableau ci-dessous présente les thèmes qui sont abordés lors de ces séances. L'ordre de ces thèmes n'est pas impératif.

Aucune connaissance relative aux modules éventuellement rencontrés lors de ces séances n'est exigible des étudiants.

Thèmes	Exemples d'activité. Commentaires.
Algorithmes élémentaires opérant par boucles simples.	Calculs de sommes et produits. Calculs des termes d'une suite récurrente (ordre 1, ordre supérieur), liste des termes, chaînes de caractères.
Algorithme opérant par boucles dans un tableau unidimensionnel.	Recherche d'un élément. Recherche du maximum, du second maximum. <i>Manipulations élémentaires d'un tableau unidimensionnel (indexation, extraction, etc.).</i>
Lecture et écriture dans un fichier texte.	
Utilisation de modules, de bibliothèques.	Calculs statistiques sur des données. Représentation graphique (histogrammes, etc.).
Algorithmes opérant par boucles imbriquées.	Recherche d'un facteur (ou d'un mot) dans un texte. Recherche des deux valeurs les plus proches dans un tableau. Manipulations élémentaires des tableaux à deux dimensions (indexation et extraction, etc.). Calculs de la somme, du produit et de la transposée d'une matrice. <i>On en profitera pour introduire la bibliothèque « NUMPY ».</i>
Recherche dichotomique.	Recherche de valeurs approchées d'une racine d'une équation algébrique. Recherche dichotomique dans un tableau trié. <i>On met en évidence une accélération du processus.</i>
Fonctions récursives.	Factorielles, suites récurrentes. Algorithme d'exponentiation rapide. Dessins de fractales. <i>On évite de se cantonner à des fonctions mathématiques.</i>
Matrices de pixels et images.	Obtention d'une image en niveaux de gris, image miroir, négatif. Algorithmes de rotation, de réduction ou d'agrandissement. Modification d'une image par convolution : flou, détection de contour, etc. <i>On pourra utiliser la bibliothèque « PIL ».</i>
Tris.	Algorithmes naïfs : tri par insertion, par sélection. Tri par comptage. Application aux statistiques : tri d'une série statistique, recherche de la médiane (éventuellement des quartiles). <i>On pourra faire le lien entre le tri par comptage et la recherche des effectifs d'apparition dans une liste.</i>

## 2 Programme du semestre 2

On approfondit, via les leçons et travaux pratiques, le travail entrepris au premier semestre concernant la discipline et les méthodes de programmation.

### 2.1 Méthodes de programmation et dictionnaire

Même si on ne parle pas de preuve d'algorithme, on insistera sur l'importance des tests dans la mise au point des programmes.

Notions	Exemples d'activité. Commentaires.
Identifiants et valeurs. Objets mutables et non mutables, portée d'un identifiant, effets de bord.	<i>On mettra en évidence le phénomène d'aliasing et son impact dans le cas d'objets mutables (listes).</i>
Dictionnaires, clés et valeurs. Usage des dictionnaires en programmation Python. Syntaxe pour l'écriture des dictionnaires. Parcours d'un dictionnaire.	Nombre d'éléments distincts dans une liste, construction d'un index.

### 2.2 Bases de données

On se limite volontairement à une description applicative des bases de données en langage SQL. Il s'agit de permettre d'interroger une base présentant des données à travers plusieurs relations.

Notions	Exemples d'activité. Commentaires.
Vocabulaire des bases de données : tables ou relations, attributs ou colonnes, domaine, schéma de tables, enregistrements ou lignes, types de données.	<i>On présente ces concepts à travers de nombreux exemples. On s'en tient à une notion sommaire de domaine : entier, flottant, chaîne; aucune considération quant aux types des moteurs SQL n'est au programme. Aucune notion relative à la représentation des dates n'est au programme; en tant que de besoin on s'appuie sur des types numériques ou chaîne pour lesquels la relation d'ordre coïncide avec l'écoulement du temps. Toute notion relative aux collations est hors programme; en tant que de besoin on se place dans l'hypothèse que la relation d'ordre correspond à l'ordre lexicographique usuel. NULL est hors programme.</i>
Clé primaire, clé étrangère	<i>On se limite au cas où une clé primaire est associée à un unique attribut.</i>
Requêtes SELECT avec simple clause WHERE (sélection), projection, renommage AS.  Utilisation des mots-clés DISTINCT et ORDER BY.	<i>Les opérateurs au programme sont +, -, *, / (on passe outre les subtilités liées à la division entière ou flottante), =, &lt;&gt;, &lt;, &lt;=, &gt;, &gt;=, AND, OR, NOT. D'autres mots-clés comme OFFSET et LIMIT pourront être utilisés mais leur maîtrise n'est pas au programme.</i>
Jointures $T_1 \text{ JOIN } T_2 \dots \text{ JOIN } T_n$ ON $\phi$ .	<i>On présente les jointures en lien avec la notion de relations entre tables. On se limite aux équi-jointures : <math>\phi</math> est une conjonction d'égalités.</i>
Agrégation avec les fonctions MIN, MAX, SUM, AVG et COUNT, y compris avec GROUP BY.	<i>Pour la mise en œuvre des agrégats, on s'en tient à la norme SQL99. Les requêtes imbriquées ne sont pas au programme.</i>

### Mise en œuvre

La création de tables et la suppression de tables au travers du langage SQL sont hors programme. La mise en œuvre effective se fait au travers d'un logiciel permettant d'interroger une base de données à l'aide de requêtes SQL comme *MySQL* ou *SQLite*. Récupérer le résultat d'une requête à partir d'un programme n'est pas un objectif.

Sont hors programme : la notion de modèle logique *vs* physique, les bases de données non relationnelles, les méthodes de modélisation de base, les fragments DDL, TCL et ACL du langage SQL, les transactions, l'optimisation de requêtes par l'algèbre relationnelle.

## 2.3 Graphes

Il s'agit de définir le modèle des graphes, leurs représentations et leurs manipulations.

On s'efforce de mettre en avant des applications importantes et si possibles modernes : réseau de transport, graphe du web, réseaux sociaux, bio-informatique. On précise autant que possible la taille typique de tels graphes.

Notions	Exemples d'activité. Commentaires.
Vocabulaire des graphes. Graphe orienté, graphe non orienté. Sommet (ou nœud); arc, arête. Boucle. Chemin d'un sommet à un autre. Connexité dans les graphes non orientés. Matrice d'adjacence. Graphe $G = (S, A)$ .	<i>On présente l'implémentation des graphes à l'aide de listes d'adjacence (rassemblées par exemple dans une liste ou dans un dictionnaire). On n'évoque ni multi-arcs ni multi-arêtes.</i>
Pondération d'un graphe. Étiquettes des arcs ou des arêtes d'un graphe.	<i>On motive l'ajout d'information à un graphe par des exemples concrets.</i>
Parcours d'un graphe. Parcours en largeur.	<i>La file sera représentée par une liste. On pourra évoquer le problème de cette représentation naïve en terme d'efficacité mais aucune connaissance sur d'autres représentations plus performante n'est au programme.</i>

## 2.4 Méthodes numériques

Cette section propose des sujets de travaux pratiques venant compléter le programme du deuxième semestre.

Thèmes	Exemples d'activité. Commentaires.
Méthode des rectangles.	Comparaison avec d'autres méthodes : méthode de Newton, méthode des trapèzes.
Simulation de variables aléatoires suivant des lois usuelles : Bernoulli, binomiale, uniforme. Estimation d'une probabilité, estimation de l'espérance et de la variance.	Simulation d'expériences et de variables aléatoires. Simulation d'une variable aléatoire à l'aide de sa fonction de répartition. <i>La justification de cette estimation sera donnée en deuxième année</i>

### 3 Programme des semestres 3 et 4

Le programme de deuxième année est constitué de séances de travaux pratiques qui poursuivent les objectifs suivants :

- approfondir les connaissances acquises au deuxième semestre notamment sur les bases de données et sur les graphes;
- programmer des méthodes numériques vues en mathématiques;
- mettre en application des notions vues en statistiques et probabilités dont les tests statistiques;
- programmer de nouveaux algorithmes sur des applications en lien avec d'autres disciplines, notamment la biologie.

Comme au premier semestre, les tableaux ci-dessous présentent les thèmes qui sont abordés lors de ces séances. L'ordre de ces thèmes n'est pas impératif.

Aucune connaissance relative aux modules éventuellement rencontrés lors de ces séances n'est exigible des étudiants.

Les deux parties du programme représentent un volume horaire équivalent.

#### 3.1 Méthodes numériques et statistiques

Thèmes	Exemples d'activités Commentaires
Méthode d'Euler explicite pour la résolution approchée d'équation différentielle ordinaire d'ordre 1.	Modèle logistique, comparaison avec d'autres méthodes de résolution approchée (méthode de Heun par exemple). <i>L'impact du pas de discrétisation sur la qualité des résultats et sur le temps de calcul est mis en évidence.</i> <i>Les résultats obtenus peuvent être comparés avec une résolution exacte ou avec une fonction de résolution approchée fournie par un module.</i>
Simulation d'une variable aléatoire de loi géométrique à l'aide de la loi de Bernoulli.  Estimation de l'espérance, de la loi d'une variable aléatoire à partir de simulation.	On pourra simuler des variables aléatoires à l'aide de la loi uniforme et de la réciproque de la fonction de répartition : cas discret et continu. Exemples de la loi de Poisson, de la loi exponentielle et de la loi normale. Exemples de chaînes de Markov. Fonction de répartition empirique.
Illustration numérique de convergence en lois. Simulation d'une variable aléatoire suivant une loi de Poisson à l'aide d'une loi binomiale. Intervalle de confiance pour le paramètre d'une loi de Bernoulli.	Illustration du théorème central limite.  <i>La notion théorique d'intervalle de confiance n'est pas au programme</i>
Approfondissement sur les statistiques : simulation et mécanisme de tests. Simulation de variables aléatoires de lois de khi-2 et de Student. Elaboration d'un test de moyenne, dit de Student, sur un petit échantillon gaussien.	Elaboration d'un test de la moyenne, cas particulier d'une proportion. On affichera les histogrammes obtenus. On pourra visualiser la convergence des lois de Student vers une loi normale. <i>La connaissance des lois de Student et du khi-2 n'est pas un attendu du programme. On pourra utiliser les simulateurs intégrés dans Python. La mise en place de test statistique (hormis le test de conformité à la moyenne) doit être accompagné, les outils mathématiques sous-jacents ne sont pas au programme</i>

On pourra compléter cette partie par une étude sur un autre sujet ou approfondir un des thèmes précédents. Les thèmes suivants ont été choisis par leur lien possibles avec des applications possible en SVT mais ils ne

sont que des exemples proposés et ne sont pas des attendus exigibles du programme.

Exemples de thèmes libres	Exemples d'activités Commentaires
Complément sur la méthode d'Euler.	Méthode d'Euler explicite pour la résolution approchée de système d'équations différentielles d'ordre 1 et d'équation différentielle d'ordre 2. Modèle de Lotka-Volterra, modèle SIR, système différentiel issu d'une équation de cinétique chimique, équation d'un oscillateur harmonique amorti. <i>Une programmation vectorisée, même si elle peut être proposée, n'est pas un attendu du programme.</i>
Méthode du pivot de Gauss.	Résolution d'un système linéaire. <i>Une version sans recherche de pivot peut être proposée en premier puis une recherche de pivot partiel peut être programmée.</i> <i>On pourra mettre en évidence l'impact de la taille du système sur le temps de calcul.</i> <i>La programmation sans aide n'est pas un attendu du programme.</i>
Autour des équations aux dérivées partielles.	Équation de la chaleur, couplage réaction-diffusion. <i>Aucune connaissance sur les équations aux dérivées partielles n'est exigible.</i>

### 3.2 Approfondissements des concepts informatiques

Thèmes	Exemples d'activités Commentaires
Révisions et approfondissements sur les graphes. Parcours en profondeur. Plus court chemin dans un graphe pondéré.	Algorithme de Dijkstra. Problème du voyageur de commerce. Algorithme glouton, colonie de fourmis, recuit simulé. <i>L'objectif est de reprendre la structure de graphe à travers des approfondissements.</i>
Révisions des tris naïfs et exemple de tri récursif.	Tri fusion ou tri rapide. On pourra visualiser à l'aide de graphique l'accélération du temps de calcul. <i>La programmation du tri rapide en place n'est pas un objectif du programme. L'objectif sur les deux années est qu'un étudiant sache programmer un tri de son choix de façon autonome.</i>
Révisions et approfondissements sur les bases de données : révision sur les jointures et agrégations	<i>A partir d'une base de données comprenant 3 ou 4 tables ou relations, on approfondit la notion de jointure interne en lien avec la notion d'associations entre entités.</i> <i>Pour la notion d'agrégation, on présente quelques exemples de requêtes imbriquées.</i> <i>On marque la différence entre WHERE et HAVING sur des exemples.</i>

L'objectif de cette dernière partie est à travers l'étude d'un thème, sur plusieurs séances, de permettre aux étudiants de gagner en autonomie (choix de la représentation des données, choix du découpage en fonctions, etc).

Les thèmes suivants ont été choisis par leurs applications possibles en lien avec la SVT mais ils ne sont que des exemples proposés et ne sont pas des attendus exigibles du programme.

Exemples de thèmes libres	Exemples d'activités <i>Commentaires</i>
Informatique pour la génétique.	Recherche de motif : recherche exacte d'une ou plusieurs séquences courtes dans un génome. Algorithme naïf, algorithme de Rabin-Karp. Alignement de séquences en présence d'erreurs de séquençage ou de mutations. Distance d'édition, algorithmes de Needleman-Wunsch, de Smith-Waterman. Construction d'arbres phylogénétiques.
Images.	Segmentation d'images : comment partitionner une image en zones connexes? <i>La structure union-find peut être introduite dans ce but.</i>
Apprentissage et classification.	Algorithme des k-moyennes. Classification par $k$ plus proches voisins. <i>On pourra travailler autant sur des données réelles que sur des données issues de simulations (clusters gaussiens par exemple).</i>

# A Langage Python

Cette annexe liste limitativement les éléments du langage Python (version 3 ou supérieure) dont la connaissance est exigible des étudiants. Aucun concept sous-jacent n'est exigible au titre de la présente annexe.

Aucune connaissance sur un module particulier n'est exigible des étudiants.

Toute utilisation d'autres éléments du langage que ceux que liste cette annexe, ou d'une fonction d'un module, doit obligatoirement être accompagnée de la documentation utile, sans que puisse être attendue une quelconque maîtrise par les étudiants de ces éléments.

## Traits généraux

- Principe d'indentation.
- Portée lexicale : lorsqu'une expression fait référence à une variable à l'intérieur d'une fonction, Python cherche la valeur définie à l'intérieur de la fonction et à défaut la valeur dans l'espace global du module.

## Types de base

- Opérations sur les entiers (`int`) : `+`, `-`, `*`, `**`, avec des opérandes positifs.
- Opérations sur les flottants (`float`) : `+`, `-`, `*`, `/`, `**`.
- Opérations sur les booléens (`bool`) : `not`, `or`, `and`.
- Comparaisons `==`, `!=`, `<`, `>`, `<=`, `>=`.

## Types structurés

- Structures indicées immuables (chaînes de caractères) : `len`, accès par indice positif valide, concaténation `+`, répétition `*`, tranche.
- Listes : création par compréhension `[e for x in s]`, par `append` successifs; `len`, accès par indice positif valide; concaténation `+`, répétition `*`, tranche, copie; `pop` en dernière position.
- Dictionnaires : création, accès, insertion, `len`, `copy`.

## Structures de contrôle

- Instruction d'affectation avec `=`.
- Instruction conditionnelle : `if`, `elif`, `else`.
- Boucle `while` (sans `else`), `return` dans un corps de boucle.
- Boucle `for` (sans `else`) et itération sur `range(a, b)`, une chaîne de caractères une liste, un dictionnaire au travers des méthodes `keys` et `items`.
- Définition d'une fonction `def f(p1, ..., pn), return`.

## Divers

- Introduction d'un commentaire avec `#`.
- Utilisation simple de `print`, sans paramètre facultatif.
- Importation de modules avec `import module`, `import module as alias`, `from module import f, g, ...`



# **Classes préparatoires aux grandes écoles**

## **Filière scientifique**

### **Voie Biologie, chimie, physique et sciences de la Terre (BCPST)**

#### **Annexe 5**

#### **Programme de géographie 2<sup>nd</sup>e année**

## PROGRAMME DE GEOGRAPHIE POUR LES CLASSES PREPARATOIRES DE LA VOIE BCPST

### Environnement et aménagement durable des territoires ruraux et périurbains en France métropolitaine et d'outre-mer

#### I. - OBJECTIFS DE FORMATION

La géographie vise à la compréhension d'un monde complexe que l'on ne peut appréhender sans une connaissance fine des acteurs de l'aménagement des territoires et sans une approche multiscalaire des problématiques écologiques, économiques, sociétales et culturelles.

Ce programme doit permettre aux étudiants de pratiquer la géographie comme une discipline d'ouverture vers les sciences humaines et d'identification des enjeux sociétaux. L'approche géographique partage les acquis des sciences de la vie et des sciences de la Terre de façon à croiser les regards autour de problématiques (ressources, risques...), d'objets d'étude (paysage, territoire), de documents et d'outils (cartes, images).

L'ambition du programme est de sensibiliser les futurs agronomes et vétérinaires à des problématiques géographiques liées aux espaces ruraux et périurbains du territoire français. Ce programme constitue un premier jalon vers des compétences développées au cours des formations d'agronomes ou de vétérinaires, en particulier la capacité à établir un état des lieux d'une réalité et de ses problèmes ainsi que la capacité à accompagner divers acteurs du monde rural (personnes, organisations). L'intégration des dimensions écologiques, économiques, sociétales et culturelles permettra aux futurs agronomes et vétérinaires d'appréhender les espaces ruraux et périurbains dans leur complexité et d'inscrire le concept de durabilité au cœur de leurs questionnements et de leurs actions futures.

La maîtrise de la lecture de la carte, placée au centre de l'épreuve, permettra de révéler de manière appliquée des complémentarités, des conflits, des contradictions perçus trop souvent de manière théorique et fragmentée. Elle constituera en outre le support d'une approche multiscalaire des réalités territoriales.

#### II. - COMPETENCES A METTRE EN ŒUVRE DANS LE CADRE DE L'ENSEIGNEMENT DE LA GEOGRAPHIE

<b>Compétences spécifiques en géographie</b>	<b>Se repérer dans l'espace</b>	<ul style="list-style-type: none"><li>- Localiser une carte</li><li>- Situer la carte dans des logiques territoriales plus globales</li><li>- Raisonner à différentes échelles</li></ul>
	<b>Analyser et comprendre un corpus documentaire en fonction d'un sujet donné</b>	<ul style="list-style-type: none"><li>- Lire et analyser une carte topographique</li><li>- Analyser et confronter des documents de nature différente (images, données statistiques, affiches publicitaires ...)</li></ul>
	<b>Sélectionner l'information</b>	<ul style="list-style-type: none"><li>- Exploiter et confronter des informations en fonction d'une thématique géographique appliquée à un territoire</li><li>- Appliquer au territoire d'étude les</li></ul>

		<p>informations tirées des documents et/ou des connaissances personnelles</p> <p>Porter un regard critique sur les documents (sources, contexte, indicateurs choisis, échelles)</p>
	<b>Raisonnement, justifier une démarche et des choix</b>	<ul style="list-style-type: none"> <li>- Mettre en perspective les documents pour faire émerger des enjeux territoriaux</li> <li>- Formuler, vérifier des hypothèses sur les logiques spatiales propres à un territoire</li> <li>- Mobiliser des connaissances générales pour expliquer une situation locale</li> </ul>
	<b>S'informer et appuyer son raisonnement à partir de l'outil numérique</b>	<ul style="list-style-type: none"> <li>- Utiliser des sites d'informations et de recherche en géographie</li> <li>- Exploiter les informations tirées d'un SIG</li> </ul>
	<b>Confronter le savoir acquis en géographie avec d'autres matières</b>	<ul style="list-style-type: none"> <li>- Mettre en relation les connaissances géographiques avec celles des autres matières</li> <li>- Utiliser les démarches de raisonnement communes à différentes matières et les appliquer sur des outils de travail communs (outil numérique, données satellitaires, sortie de terrain...)</li> </ul>
<b>Compétences générales</b>	<b>Compétences de communication orale</b>	<ul style="list-style-type: none"> <li>- Maîtriser un ensemble documentaire inconnu</li> <li>- Argumenter</li> <li>- Dialoguer, défendre son point de vue</li> <li>- Compléter, nuancer ou corriger son propos initial à la faveur d'un entretien oral</li> </ul>
	<b>Compétences mobilisables en situation professionnelle</b>	<ul style="list-style-type: none"> <li>- Sélectionner, décrire et analyser des données physiques, économiques, sociales et environnementales et les confronter à des connaissances générales</li> <li>- Evaluer les contraintes et ressources d'un territoire au regard d'une question donnée</li> <li>- Identifier les acteurs du monde rural (élus, professionnels, associations, usagers, etc.) et comprendre leurs logiques, y compris lorsqu'elles révèlent des contradictions</li> <li>- Envisager différentes solutions et pistes de développement réalistes</li> </ul>

### III. – ESPACES ET AXES DU PROGRAMME

#### III. 1 – Les espaces du programme

Le programme ancre la réflexion dans la diversité des territoires de la France métropolitaine et de la France d'outre-mer (limitée aux DROM). Les espaces ruraux et périurbains sont à étudier dans leurs diverses composantes et relations. Les espaces urbains présents sur la carte ne sont à considérer qu'en fonction de leurs relations avec les espaces ruraux et périurbains.

#### III. 2 – Axes et thématiques transversales du programme

Trois axes constituent les points d'entrée principaux dans le programme.

##### - Environnement et milieux : ressources, gestion, protection

Il s'agit de saisir la diversité et la complexité des milieux dans leurs composantes naturelles et anthropiques au prisme des pratiques, des représentations et des politiques associées à ces espaces.

##### - Espaces productifs et dynamiques socio-économiques

Dans un contexte de profondes mutations économiques et sociales pour partie liées aux processus de mondialisation, sont étudiés les filières productives et les acteurs qui structurent les territoires ruraux et périurbains.

Au-delà de ces seules dimensions productives, c'est bien la multifonctionnalité des espaces ruraux et périurbains qu'il conviendra d'examiner.

Une attention toute particulière sera portée aux enjeux résidentiels nouveaux et différenciés des espaces ruraux et périurbains, et à leurs impacts économiques, sociaux et culturels sur les territoires.

##### - Aménagement des territoires

Les enjeux du développement durable au sein des territoires ruraux et périurbains deviennent une préoccupation majeure des politiques d'aménagement. Confrontés à de nouvelles demandes et à de nouveaux cadres institutionnels, les acteurs, publics et privés, la société civile et le milieu associatif, apportent des réponses variées qu'il convient d'examiner.

Afin de mettre en relation ces axes principaux, certaines **thématiques transversales** pourront servir de points de réflexion et d'articulation.

- **Les représentations** : les acteurs agissent en fonction des représentations qu'ils se font de l'espace. Véritables moteurs de l'action, elles sont décisives dans l'analyse des dynamiques territoriales rurales et périurbaines.
- **Les conflits d'usage** : les acteurs, par leurs usages, leurs représentations distinctes des espaces peuvent entrer en confrontation sur la gestion/préservation des milieux, l'exploitation économique ou l'aménagement des territoires.
- **Le patrimoine** : notion-clé dans la mise en valeur des espaces ruraux, le patrimoine est un construit dépendant des représentations différentes, parfois conflictuelles des divers acteurs. Protection, exploitation, classement... font de la patrimonialisation des territoires une pratique centrale.
- **La question alimentaire** : dans le cadre d'une formation agronomique et vétérinaire, le rapport à l'alimentation fera l'objet d'une attention particulière. Seront abordés les enjeux de la question animale : maintien de la biodiversité, place de l'animal dans les différents modèles agricoles, mais

aussi conflits territoriaux centrés sur l'animal (loup, ours). Les questions liées aux productions végétales à destination alimentaire pourront de même être saisies dans une approche géographique : conséquences de la diffusion massive de certaines cultures, controverses sur les pratiques agricoles et leurs impacts, etc...

#### **IV. - LES DOCUMENTS ET OUTILS**

**Ce programme doit être abordé avec des documents** qui permettent de diversifier les approches thématiques et régionales.

##### **IV.1 La carte topographique**

La carte topographique au 1/25000<sup>ème</sup> est le document de base. A l'issue de leur année de formation, les étudiants doivent être en mesure de savoir la lire avec précision et d'en proposer un commentaire problématisé.

Cet exercice sur la carte topographique suppose des candidats une capacité à mobiliser un vocabulaire géographique sur l'ensemble des axes du programme. A cet égard, le vocabulaire élémentaire de géographie physique doit être connu (grandes notions de géomorphologie, principales formations végétales et caractéristiques climatiques) et mobilisé à bon escient en fonction du thème. Si la maîtrise de ce vocabulaire semble nécessaire pour un commentaire de carte topographique, on ne peut en aucun cas exiger des candidats qu'ils expliquent les processus de mise en place de ces éléments, en particulier géomorphologiques.

##### **IV. 2 Les documents complémentaires**

L'analyse de la carte s'appuie sur d'autres types de documents. Parmi eux peuvent figurer : des études statistiques, des photos et images, des textes, des documents de promotion territoriale ou économique, des cartes thématiques (plan de prévention des risques, zonages d'un EPCI ou d'une autre collectivité territoriale...). Les sources peuvent être variées : institutionnelles, scientifiques, journalistiques, etc...

Parmi ces documents complémentaires, le numérique occupe une place de choix. En effet, cet outil, de manière spécifique à l'enseignement de la géographie mais également par des usages transversaux avec d'autres enseignements de la filière, se doit d'être intégré à la formation des futurs agronomes et vétérinaires. L'utilisation d'un SIG simple, disponible et gratuit comme le site du Géoportail est à privilégier.

L'utilisation du Géoportail est généralisée : chaque sujet proposé le jour de l'épreuve comprend au minimum un document issu de la banque de données du site. Dans ce cadre, les étudiants devront être familiarisés prioritairement à l'usage des couches d'information suivantes : cartes topographiques d'échelles et de dates différentes, photographies aériennes de dates différentes, carte géologique, recensement général des productions (ancien et récent), carte d'occupation des sols (Corine Land and Cover 2018), carte forestière, cartes de risques. Ces couches seront accompagnées le jour du concours de leur légende et des informations nécessaires à leur compréhension (date, métadonnées, source...). De plus, il est attendu que l'étudiant sache proposer et justifier l'utilisation d'une ou plusieurs couches complémentaires pour enrichir son analyse.