

PROGRAMME DE SCIENCES INDUSTRIELLES DE L'INGÉNIEUR DANS LA FILIÈRE TECHNOLOGIE ET SCIENCES INDUSTRIELLES

1. Objectifs de formation

1.1. Finalité

Le programme de sciences industrielles de l'ingénieur (SII) de la filière TSI s'inscrit dans un parcours de formation initiale pour accéder au titre d'ingénieur. Il trouve ses racines dans l'enseignement de spécialité du cycle terminal en Sciences et Techniques de l'Industrie et du Développement Durable. L'objectif de ce programme est de proposer des contenus d'enseignements qui permettent de développer progressivement les compétences nécessaires à l'intégration dans une grande école et à l'exercice des métiers d'ingénieurs. Ce programme est ambitieux quant au développement de compétences scientifiques et technologiques qui soutiennent l'expertise du futur ingénieur. Il l'est aussi pour le développement de compétences transversales nécessaires pour communiquer, travailler en équipe, exercer un sens critique et des responsabilités de manière éthique et déontologique. En cohérence avec les objectifs du cycle initial de la formation aux métiers de l'ingénierie, ce programme contribue à l'approche pédagogique par les STEM (*Science, Technology, Engineering and Mathematics*).

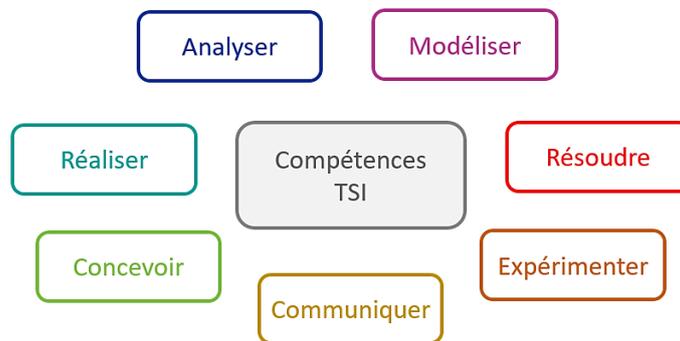
1.2. Objectifs généraux

Les ingénieurs doivent être en capacité de résoudre de façon innovante des problèmes inédits afin de répondre aux besoins des personnes et d'apporter un progrès dans leur qualité de vie. Ils participent aux processus de développement des systèmes à chaque étape de leur cycle de vie, de la caractérisation du besoin jusqu'au recyclage, en respectant les contraintes de développement durable et d'écoconception.

Cette capacité des ingénieurs à proposer des solutions innovantes est plus que jamais indispensable au développement d'une industrie capable de faire face aux grands enjeux sociétaux, économiques et environnementaux. Ces enjeux sont notamment ceux de la transition énergétique, la préservation de la qualité de l'environnement, la progression des technologies du numérique, la mutation des métropoles et des territoires, l'évolution des besoins alimentaires et des exigences en matière de santé pour des humains toujours plus nombreux sur notre planète. Dans un contexte de concurrence mondialisée, la capacité d'innovation des ingénieurs est nécessaire à l'industrie de notre pays qui doit demeurer compétitive et souveraine.

Les objectifs généraux du programme de SII en CPGE TSI visent à développer les compétences clés dans le large domaine des sciences industrielles de l'ingénieur qui sont nécessaires à l'exercice du métier d'ingénieur. Celles-ci sont consolidées et complétées par la formation poursuivie jusqu'à l'obtention du titre d'ingénieur.

L'enseignement des SII en CPGE TSI a également pour objectif d'apporter aux étudiants des méthodes et des outils qui leur permettront de s'adapter aux évolutions permanentes des sciences et des technologies et de communiquer avec l'ensemble des acteurs associés à l'exercice des métiers d'ingénieurs et scientifiques.

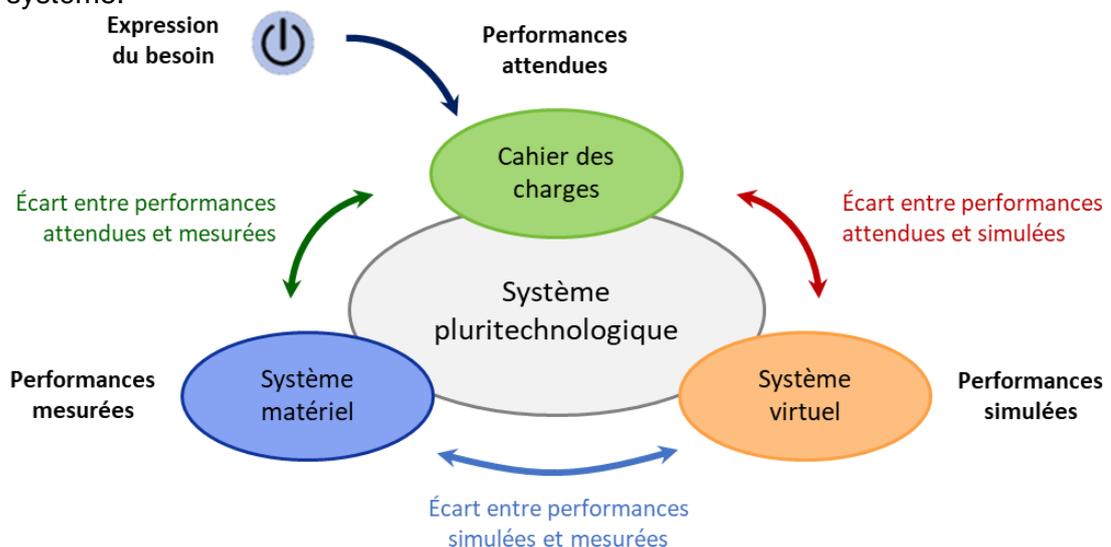


Les compétences générales de l'ingénieur développées en TSI

1.3. La démarche des enseignements en CPGE TSI

L'approche pédagogique et didactique des enseignements en TSI s'organise autour de systèmes pluritechnologiques. Chaque système est défini à partir de besoins fonctionnels et d'exigences, de modèles numériques et d'un système matériel. Un système sera étudié dans sa globalité à partir de ces trois approches imbriquées :

- la réalité du besoin ou exigences fonctionnelles. Elle se décline dans le cahier des charges défini avec un client ;
- la réalité virtuelle d'un système. Elle se traduit dans l'élaboration d'un modèle permettant de simuler son comportement afin d'en prévoir et d'en évaluer les performances, et de valider les organisations fonctionnelle et structurelle ;
- la réalité matérielle d'un système. Elle se traduit par la réalisation d'un prototype fonctionnel qui permet de valider par expérimentation les performances du produit ou système.



La démarche pédagogique et didactique en sciences industrielles de l'ingénieur

Les objets et les systèmes, dans leur complexité, mobilisent plusieurs formes d'énergie et sont communicants. Ils sont pluritechnologiques.

La démarche en sciences industrielles de l'ingénieur en TSI vise à :

- contribuer à l'élaboration des trois réalités du système pluritechnologique (le cahier des charges, le système virtuel et le système matériel) ;
- comparer les performances issues de ces trois réalités ;
- optimiser le système virtuel et le système matériel afin de faire converger leurs performances vers celles attendues au cahier des charges.

Les contenus du programme de TSI permettent aux étudiants d'investir complètement la démarche de l'ingénieur en s'intéressant à toutes les représentations des systèmes. Pour cela les enseignements en TSI installent progressivement l'ensemble des connaissances et des compétences nécessaires à la maîtrise des différentes représentations d'un même objet ou système, à la comparaison des différentes performances, à l'optimisation des systèmes dans leurs réalités numérique et matérielle, afin de répondre aux attentes du client.

À partir de l'analyse du cahier des charges, des solutions innovantes sont conçues et réalisées. La réalisation consiste en l'élaboration de modèles à l'aide d'outils numériques et de prototypes matériels de systèmes intégrant la chaîne de puissance et la chaîne d'information, capables de valider tout ou partie du cahier des charges.

1.4. Usage de la liberté pédagogique

Le programme définit les obligations faites aux professeurs des contenus à enseigner, les mêmes pour tous les étudiants, garantes de l'équité d'une formation offrant à chacun les mêmes chances de réussite. Les finalités et objectifs généraux de la formation en sciences industrielles de l'ingénieur laissent aux enseignants le choix pédagogique de l'organisation des enseignements et de ses méthodes. La nature des enseignements en sciences industrielles de l'ingénieur suppose la mise en œuvre d'une didactique naturellement liée à la discipline qui impose une réflexion sur le développement des compétences, la transmission des connaissances et leur ordonnancement dans la programmation des apprentissages.

La pédagogie mise en œuvre valorise et s'appuie sur les compétences acquises par les élèves de la voie technologique et du développement durable par la mise en application concrète de connaissances scientifiques et technologiques sur des supports d'enseignement représentatifs de solutions innovantes. Les solutions contemporaines sont mises en perspective avec l'histoire des sociétés, des technologies, avec les préoccupations de respect de l'environnement et des ressources naturelles, de façon à construire les bases d'une culture d'ingénieur éthique et responsable.

2. Programme

Le programme est organisé en sept compétences générales déclinées en compétences attendues qui pourront être évaluées en fin de cycle.

Partant de ces indications de fin de cycle, le programme détaille les compétences développées, précise les connaissances associées et fournit un indicateur de positionnement temporel dans le cycle.

Les compétences développées et les connaissances associées sont positionnées dans les semestres, cela signifie :

- qu'elles doivent être acquises en fin du semestre précisé ;
- qu'elles ont pu être introduites au cours des semestres précédents ;
- qu'elles peuvent être mobilisées aux semestres suivants.

Les compétences générales et compétences attendues sont détaillées ci-dessous.

A – Analyser

- A1 – Analyser le besoin et les exigences
- A2 – Définir les frontières de l'analyse
- A3 – Analyser l'organisation fonctionnelle et structurelle
- A4 – Analyser les performances et les écarts
- A5 – Analyser un compromis produit-procédés-matériaux

B – Modéliser

- B1 – Identifier les phénomènes physiques pour les modéliser et caractériser les grandeurs nécessaires
- B2 – Proposer un modèle de connaissance et de comportement
- B3 – Valider un modèle

C – Résoudre

- C1 – Proposer une démarche de résolution
- C2 – Mettre en œuvre une démarche de résolution analytique
- C3 – Mettre en œuvre une démarche de résolution numérique

D – Expérimenter

- D1 – Découvrir le fonctionnement d'un système pluritechnologique et le mettre en œuvre
- D2 – Proposer et justifier un protocole expérimental
- D3 – Mettre en œuvre un protocole expérimental

E – Communiquer

- E1 – Rechercher et traiter des informations
- E2 – Produire et échanger de l'information

F – Concevoir

- F1 – Écoconcevoir l'architecture d'un système innovant
- F2 – Proposer et choisir des solutions techniques
- F3 – Dimensionner une solution technique choisie dans une démarche de développement durable

G – Réaliser

- G1 – Réaliser tout ou partie d'un prototype

Les liens avec l'enseignement d'informatique du tronc commun sont identifiés par le symbole $\Leftrightarrow I$.

A – Analyser

A1 – Analyser le besoin et les exigences

Compétences développées	Connaissances associées	Semestre
Décrire le besoin et les exigences.	Ingénierie Système et diagrammes associés.	S1
<p><i>Commentaire</i></p> <p><i>Les diagrammes d'Ingénierie Système - SysML (uc, req) peuvent être proposés à lire, à compléter ou à créer en s'appuyant sur un document fourni présentant la syntaxe.</i></p>		
Traduire un besoin fonctionnel en exigences.	Impact environnemental. Analyse du cycle de vie (extraction, fabrication, utilisation, fin de vie, recyclage et transport). Multi-étapes et multicritères.	S1
Définir les domaines d'application et les critères technico-économiques et environnementaux.		
Qualifier et quantifier les exigences.		
Évaluer l'impact environnemental et sociétal.		
<p><i>Commentaires</i></p> <p><i>Il s'agit de prendre en compte les exigences liées au développement durable et de sensibiliser aux aspects sociétaux.</i></p> <p><i>L'évaluation de l'impact environnemental et sociétal s'appuie sur les 3 piliers du développement durable (social, économique et environnemental).</i></p>		

A2 – Définir les frontières de l'analyse

Compétences développées	Connaissances associées	Semestre
Isoler un système et justifier l'isolement.	Frontière de l'étude.	S2
Définir les éléments influents du milieu extérieur.	Milieu extérieur.	
Identifier la nature des flux échangés traversant la frontière d'étude.	Flux de matière, d'énergie et d'information.	S2
<p><i>Commentaire</i></p> <p><i>Les diagrammes d'Ingénierie Système - SysML (bdd, ibd) peuvent être proposés à lire, à compléter ou à créer en s'appuyant sur un document fourni présentant la syntaxe.</i></p>		

A3 – Analyser l'organisation fonctionnelle et structurelle

Compétences développées	Connaissances associées	Semestre
Associer les fonctions aux constituants.	Architecture fonctionnelle.	S1
Justifier le choix des constituants dédiés aux fonctions d'un système.	Chaînes fonctionnelles (chaîne d'information et chaîne de puissance).	S4
Identifier et décrire les chaînes fonctionnelles du système.	Fonctions acquérir, traiter, communiquer, restituer, alimenter, stocker, adapter, moduler, convertir, transmettre et agir.	S1
Identifier et décrire les liens entre les chaînes fonctionnelles.	Flux de matière, d'énergie et d'information.	S1
Identifier l'architecture structurelle d'un système.	Architecture structurelle. Diagramme de définition de blocs.	S1
Identifier la nature des flux échangés entre les différents constituants.	Diagramme de bloc interne. Flux de matière, d'énergie et d'information (définition, nature et codage).	S1
<p><i>Commentaires</i></p> <p><i>La description des chaînes fonctionnelles de différents systèmes permet de construire une culture technologique.</i></p> <p><i>Les chaînes fonctionnelles, diagrammes de définition de blocs et diagrammes de bloc interne peuvent être à lire ou à compléter avec les éléments syntaxiques fournis.</i></p>		
Caractériser un constituant de la chaîne de puissance.	Alimentation d'énergie. Association de préactionneurs et d'actionneurs : – caractéristiques ; – réversibilité ; – domaines d'application. Transmetteurs de puissance : – caractéristiques ; – réversibilité ; – domaines d'application.	S3
Caractériser un constituant de la chaîne d'information.	Capteurs : – fonctions ; – nature des grandeurs physiques d'entrée et de sortie ; – nature du signal et support de l'information. Carte programmable.	S2
<p><i>Commentaire</i></p> <p><i>Les étudiants doivent être capables de justifier un choix qualitatif de constituants de la chaîne de puissance ou d'information.</i></p>		

Analyser la structure d'un programme informatique. $\Leftrightarrow I$	Analyse fonctionnelle d'un programme. Définition et appel d'une fonction. Découpage fonctionnel. Spécification de fonctions.	S1
Analyser un algorithme. $\Leftrightarrow I$	Variables (type et portée). Structures algorithmiques (boucles et tests).	S3
Analyser les principes d'intelligence artificielle. $\Leftrightarrow I$	Régression et classification, apprentissage supervisé. Phases d'apprentissage et d'inférence. Modèle linéaire monovarié, k plus proches voisins.	S3
<p><i>Commentaires</i></p> <p><i>L'apprentissage non supervisé est introduit en regard de l'apprentissage supervisé mais aucune connaissance spécifique n'est exigible.</i></p> <p><i>Les réseaux de neurones sont abordés mais aucune connaissance spécifique n'est exigible.</i></p>		
Identifier les architectures matérielles et fonctionnelles d'un réseau de communication.	Caractéristiques d'un réseau (débit, dimension, robustesse et topologie). Supports de l'information. Caractéristiques d'un canal de transmission.	S2
	Multiplexage temporel et fréquentiel.	S4
Décoder une trame en vue d'analyser les différents champs et les données échangées.	Protocole, trame et champs (rôle des champs dans une trame). Adressage physique et logique d'un constituant.	S2
<p><i>Commentaire</i></p> <p><i>L'étude des réseaux de communication est focalisée sur les concepts communs aux protocoles de communication usuels.</i></p>		
Interpréter tout ou partie de l'évolution temporelle d'un système séquentiel.	Diagramme de séquence. Lignes de vie, messages asynchrones et synchrones, boucles. Diagramme d'états. État, transition, événement, condition de garde, activité et action.	S2
<p><i>Commentaires</i></p> <p><i>Les diagrammes d'Ingénierie Système - SysML (stm) peuvent être proposés à lire, à compléter ou à créer sur des cas simples, en s'appuyant sur un document fourni présentant la syntaxe. Les diagrammes de séquence ne peuvent être proposés qu'en lecture.</i></p> <p><i>L'évolution temporelle des états et des variables d'un diagramme d'états est représentée sous la forme d'un chronogramme.</i></p>		

Identifier la structure d'un système asservi.	<p>Grandeurs d'entrée et de sortie.</p> <p>Capteur, chaîne directe, chaîne de retour, commande, comparateur, consigne, correcteur et perturbation.</p> <p>Poursuite et régulation.</p>	S1
---	--	----

A4 – Analyser les performances et les écarts

Compétences développées	Connaissances associées	Semestre
Extraire un indicateur de performance pertinent à partir des exigences ou de résultats issus de l'expérimentation ou de la simulation.	<p>Ordre de grandeur.</p> <p>Homogénéité des résultats.</p> <p>Matrice de confusion (tableau de contingence), sensibilité et spécificité d'un test.</p>	S4
Caractériser les écarts entre les performances.		
Interpréter et vérifier la cohérence des résultats obtenus expérimentalement, analytiquement ou numériquement. $\Leftrightarrow I$		
Rechercher et proposer des causes aux écarts constatés.		
Analyser le type de correction nécessaire pour atteindre les performances attendues.	<p>Rejet de perturbation, erreurs, temps de réponse, bande passante, dépassement, marges de gain et de phase.</p> <p>Correcteur proportionnel, proportionnel intégral et à avance de phase.</p>	S3
Analyser les écarts entre les performances d'un prototype et les exigences.	<p>Dispositifs de contrôle.</p> <p>Scénarios de test.</p>	S3
<p><i>Commentaire</i></p> <p><i>Aucune connaissance spécifique n'est exigible en métrologie ou en compatibilité électromagnétique.</i></p>		

A5 – Analyser un compromis produit-procédés-matériaux

Compétences développées	Connaissances associées	Semestre
Justifier le choix d'un indicateur de performance.	Propriétés physiques des matériaux.	S3
Comparer qualitativement les caractéristiques physiques des matériaux.	Classes des matériaux, domaines généraux d'application.	
Justifier le choix d'un matériau et/ou d'un procédé.	Indices de performance. Diagrammes d'Ashby.	
Analyser les résultats d'une simulation numérique de procédés.	Impact environnemental Caractéristiques des procédés.	
<p><i>Commentaires</i></p> <p><i>Les propriétés mécaniques des matériaux sont caractérisées principalement grâce aux essais de traction.</i></p> <p><i>La connaissance des désignations normalisées des matériaux n'est pas au programme.</i></p> <p><i>Des bases de données et des outils logiciels associés permettent de conduire une analyse qualitative et quantitative sur les procédés et les matériaux en utilisant une démarche d'écoconception.</i></p> <p><i>L'objectif consiste en une découverte des procédés classiques de fabrication liés aux classes des matériaux.</i></p>		
Justifier le besoin fonctionnel d'une spécification.		S2
<p><i>Commentaire</i></p> <p><i>Les ajustements normalisés ne sont pas exigibles. Seule la nature des ajustements est à préciser (glissant, incertain et serré).</i></p>		

B – Modéliser

B1 – Identifier les phénomènes physiques pour les modéliser et caractériser les grandeurs nécessaires

Compétences développées	Connaissances associées	Semestre
Identifier les phénomènes physiques à modéliser.	Nature des grandeurs (effort et flux). Type de l'information.	S2
Caractériser les grandeurs associées utiles à la modélisation.		
Identifier les paramètres d'un modèle.		
Identifier les performances à évaluer.	Performances attendues. Ordres de grandeur. Paramètres prépondérants.	S3
Proposer des hypothèses simplificatrices en fonction des objectifs visés.		

B2 – Proposer un modèle de connaissance et de comportement

Compétences développées	Connaissances associées	Semestre
Choisir un modèle adapté aux performances à prévoir ou à évaluer.	Phénomènes physiques. Domaine de validité.	S4
Compléter un modèle multi-physique.	Paramètres d'un modèle.	S3
Associer un modèle aux constituants des chaînes fonctionnelles.	Grandeurs flux et effort. Sources parfaites et imparfaites.	
<p><i>Commentaires</i></p> <p><i>L'utilisation d'un logiciel de modélisation multi-physique permettant d'assembler des constituants issus d'une bibliothèque est privilégiée pour la modélisation des systèmes pluritechnologiques.</i></p> <p><i>Les modèles mis en œuvre couvrent différents domaines (électrique, mécanique, thermique, hydraulique et pneumatique).</i></p>		
Modéliser un système par schéma-blocs.	Schéma fonctionnel sous forme de blocs d'un système ou d'un constituant. Élaboration, manipulation et réduction de schéma-blocs. Fonctions de transfert.	S2
Établir un modèle de connaissance par des fonctions de transfert.	Systèmes linéaires continus et invariants : – causalité ; – modélisation par équations différentielles ; – transformée de Laplace ; – fonction de transfert ; – forme canonique ; – gain, ordre, classe, pôles et zéros.	S2
<p><i>Commentaires</i></p> <p><i>La présentation de la transformée de Laplace se limite à son énoncé et aux propriétés du calcul symbolique strictement nécessaires.</i></p> <p><i>Les théorèmes de la valeur finale, de la valeur initiale et du retard sont donnés sans démonstration.</i></p>		
Modéliser le signal d'entrée.	Signaux canoniques d'entrée : – impulsion ; – échelon ; – rampe ; – signaux sinusoïdaux.	S2
<p><i>Commentaire</i></p> <p><i>Un signal d'entrée périodique non sinusoïdal pourra être assimilé à sa série de Fourier limitée à sa composante continue et au premier harmonique.</i></p>		

Établir un modèle de comportement à partir d'une réponse temporelle ou fréquentielle. $\Leftrightarrow I$	Premier ordre, deuxième ordre, dérivateur, intégrateur, gain et retard. Paramètres caractéristiques. Allures des réponses indicielle et fréquentielle. Diagramme de Bode.	S2
Simplifier un modèle.	Linéarisation d'un modèle autour d'un point de fonctionnement. Pôles dominants et réduction de l'ordre du modèle : – principe ; – justification ; – limites.	S3
Modéliser un correcteur numérique. $\Leftrightarrow I$	Caractérisation des signaux à temps discret (échantillonnage et quantification). Modélisation par équations aux différences (équations de récurrence) d'un correcteur numérique.	S4
<p><i>Commentaires</i></p> <p><i>Les limites de modélisation d'un système à temps discret par un modèle à temps continu pourront être mises en évidence par l'augmentation de la période d'échantillonnage.</i></p> <p><i>Les transformées en z ne sont pas au programme.</i></p>		
Déterminer les caractéristiques d'un solide ou d'un ensemble de solides indéformables.	Solide indéformable : – définition ; – repère ; – équivalence solide/repère ; – volume et masse ; – centre d'inertie ; – matrice d'inertie, théorème de Huygens.	S3
<p><i>Commentaires</i></p> <p><i>Les calculs intégraux des éléments d'inertie (matrice et centre d'inertie) ne donnent pas lieu à évaluation.</i></p> <p><i>Le théorème de Huygens est présenté sous forme vectorielle mais l'utilisation pratique se limite au transport d'un seul moment d'inertie.</i></p>		

Proposer une modélisation des liaisons avec leurs caractéristiques géométriques.	Liaisons : – géométrie des contacts entre deux solides ; – liaisons parfaites ; – degrés de liberté ; – classe d'équivalence cinématique ; – liaisons normalisées entre solides, caractéristiques géométriques et repères d'expression privilégiés ; – paramètres géométriques linéaires et angulaires ; – symboles normalisés.	S1
Proposer un modèle cinématique à partir d'un système réel ou d'une maquette numérique volumique.	Graphe de liaisons. Schéma cinématique.	
<p><i>Commentaire</i></p> <p><i>Le paramétrage avec les angles d'Euler ou les angles de roulis, de tangage et de lacet est présenté, mais la maîtrise de ce paramétrage n'est pas exigée.</i></p>		
Modéliser la cinématique d'un ensemble de solides.	Vecteur position. Mouvements simple (translation et rotation) et composé. Trajectoire d'un point. Définition du vecteur vitesse et du vecteur taux de rotation. Définition du vecteur accélération. Composition des mouvements. Définition du contact ponctuel entre deux solides (roulement et glissement). Torseur cinématique (champ des vecteurs vitesse).	S1
Modéliser une action mécanique.	Modèle local (densités linéique, surfacique et volumique d'effort). Modèle global. Frottements sec (lois de Coulomb) et visqueux. Torseur des actions mécaniques transmissibles. Torseur d'une action mécanique extérieure. Torseurs couple et glisseur.	S2
Simplifier un modèle de mécanisme.	Liaisons équivalentes (approches cinématique et statique). Conditions et limites de la modélisation plane.	S2

Modifier un modèle de mécanisme afin de le rendre isostatique.	Mobilité du modèle d'un mécanisme. Degré d'hyperstatisme d'un modèle. Substitution de liaisons.	S2
<p><i>Commentaire</i></p> <p><i>L'identification des contraintes géométriques et dimensionnelles liées au degré d'hyperstatisme n'est pas exigible.</i></p>		
Associer un modèle poutre à un solide.	Hypothèses de géométrie. Fibre neutre et section droite. Torseur de cohésion.	S3
<p><i>Commentaire</i></p> <p><i>Les hypothèses de continuité, d'élasticité, d'homogénéité et d'isotropie des matériaux ainsi que les hypothèses de Navier-Bernoulli et de Barré de Saint-Venant tout comme l'hypothèse des petites perturbations (petites déformations et petits déplacements) sont présentées, mais aucune connaissance spécifique n'est exigible.</i></p>		
Paramétrer un modèle dans un logiciel de simulation par éléments finis.	Proposer ou justifier des conditions aux limites.	S4
Décrire le comportement d'un système séquentiel.	Diagramme d'états. État, transition, événement, condition de garde, activité et action.	S2
<p><i>Commentaire</i></p> <p><i>La description graphique permet de s'affranchir d'un langage de programmation spécifique.</i></p>		
Modéliser les modulateurs statiques d'énergie.	Règles d'interconnexion des sources parfaites. Bidirectionnalités des interrupteurs. Association des interrupteurs (cellule élémentaire de commutation). Caractéristiques des modulateurs : – nature des grandeurs d'entrée-sortie ; – réversibilité.	S4
<p><i>Commentaires</i></p> <p><i>La modélisation se limite à l'étude fonctionnelle des caractéristiques statiques des interrupteurs avec la mise en évidence de l'unidirectionnalité ou la bidirectionnalité en courant ou tension des interrupteurs.</i></p> <p><i>La modélisation des pertes dans les interrupteurs n'est pas exigible.</i></p> <p><i>La modélisation se limite aux fonctions de modulation continu-continu, continu-alternatif et alternatif-continu non commandée et à leurs associations.</i></p>		

<p>Modéliser un convertisseur électromécanique en régime permanent.</p>	<p>Modèle électromécanique de la machine à courant continu.</p> <p>Modèle statique monophasé de la machine synchrone (FEM induite, réactance synchrone et résistance).</p> <p>Modèle statique monophasé équivalent de la machine asynchrone (pertes fer négligées).</p> <p>Bilan des puissances d'un convertisseur électromécanique en régime permanent.</p>	<p>S4</p>
<p><i>Commentaires</i></p> <p><i>La physique des convertisseurs électromécaniques (machines électriques) n'est pas au programme.</i></p> <p><i>Seule la machine asynchrone à rotor à cage d'écurueil est modélisée (pilotage uniquement par le stator).</i></p> <p><i>Les modèles des machines alternatives sont fournis. Le comportement des machines alternatives est étudié en alimentation en fréquence fixe (ou lentement variable) utilisant les modèles linéaires continus statiques.</i></p> <p><i>Pour la machine asynchrone, seule la commande scalaire est étudiée (commande en « U/f » constant).</i></p> <p><i>Le fonctionnement des machines est qualifié en régime permanent dans les quatre quadrants.</i></p>		
<p>Modéliser la commande d'un ensemble asservi constitué du modulateur d'énergie, de la machine électrique et de sa charge.</p>	<p>Commande en couple des machines à courant continu et synchrone.</p> <p>Commande des machines en vitesse.</p> <p>Modèle en régime dynamique de la machine à courant continu.</p> <p>Modèle en régime dynamique dans le plan (d,q) de la machine synchrone.</p>	<p>S4</p>
<p><i>Commentaires</i></p> <p><i>La boucle interne de courant (couple) est présentée pour améliorer les performances de la boucle de vitesse angulaire.</i></p> <p><i>Il s'agit d'établir que la commande de la MS dans le plan (d,q) permet d'obtenir le même comportement qu'une MCC.</i></p> <p><i>Le modèle en régime dynamique de la machine synchrone dans le plan (d,q) est fourni sans démonstration.</i></p>		

B3 – Valider un modèle

Compétences développées	Connaissances associées	Semestre
Vérifier la cohérence du modèle choisi en confrontant les résultats analytiques et/ou numériques aux résultats expérimentaux.	Notions d'écart (absolu, relatif), mesure des écarts. Critères de performances.	S4
Préciser les limites de validité d'un modèle.	Point de fonctionnement. Non-linéarités (courbure, hystérésis, saturation et seuil) et retard pur.	S4
<p><i>Commentaire</i></p> <p><i>Les activités de simulation et d'expérimentation permettent de mettre en évidence les limites des modèles linéaires mais l'étude des modèles non-linéaires n'est pas au programme.</i></p>		
Modifier les paramètres et enrichir le modèle pour minimiser les écarts entre les résultats analytiques et/ou numériques et les résultats expérimentaux.	Hypothèses de modélisation. Paramètres d'un modèle. Paramètres influents et prépondérants.	S4

C – Résoudre

C1 – Proposer une démarche de résolution

Compétences développées	Connaissances associées	Semestre
Proposer une démarche permettant d'évaluer les performances des systèmes asservis.	Exigences : – stabilité (définition, marges de stabilité, amortissement et dépassement relatif) ; – précision (erreur/écart statique et erreur de traînage) ; – rapidité (temps de réponse à 5 %, bande passante et retard de traînage).	S2
Proposer une démarche de réglage d'un correcteur.	Correcteurs proportionnels.	S3
Choisir une démarche de résolution d'un problème d'ingénierie numérique. $\hookrightarrow I$	Décomposition d'un problème complexe en sous problèmes simples. Choix des algorithmes d'intelligence artificielle : – k plus proches voisins ; – régression linéaire monovariante.	S3
<p><i>Commentaire</i></p> <p><i>Les réseaux de neurones peuvent être évoqués mais aucune connaissance spécifique n'est exigible.</i></p>		

Proposer une démarche permettant d'obtenir une loi entrée-sortie géométrique ou cinématique. $\Leftrightarrow I$	Fermetures géométriques. Fermetures cinématiques.	S1
Proposer une démarche permettant la détermination d'une action mécanique inconnue ou d'une loi de mouvement.	Graphe de structure. Choix des isolements. Choix des équations pertinentes vis-à-vis de l'objectif. Principe Fondamental de la Statique dans un référentiel galiléen. Principe Fondamental de la Dynamique. Théorème de l'énergie cinétique.	S3
<p><i>Commentaire</i></p> <p><i>La détermination du moment dynamique peut être introduite sur quelques cas simples mais sa maîtrise ne sera ni exigible ni évaluée. La détermination des actions mécaniques inconnues peut être menée par l'usage d'un outil de simulation numérique.</i></p>		
Proposer une démarche permettant de déterminer des grandeurs électriques.	Lois de Kirchhoff. Théorème de superposition. Puissance électrique.	S1
Proposer une démarche permettant de déterminer les contraintes et/ou les déplacements le long d'une poutre.	Tronçons. Méthode des coupures.	S3
<p><i>Commentaire</i></p> <p><i>Les méthodes de résolution des problèmes hyperstatiques en résistance des matériaux ne sont pas au programme et les sollicitations ne sont pas combinées.</i></p>		

C2 – Mettre en œuvre une démarche de résolution analytique

Compétences développées	Connaissances associées	Semestre
Déterminer la réponse temporelle.	Expression de la solution de l'équation différentielle associée à un système d'ordre 1 soumis à une entrée échelon. Allure des représentations temporelles des solutions des équations différentielles d'ordre 1 et 2 pour les entrées de type échelon et sinus (en régime permanent).	S3
<p><i>Commentaires</i></p> <p><i>Le calcul des transformées de Laplace et de leurs inverses n'est pas exigé. Elles sont utilisées à partir de résultats fournis.</i></p> <p><i>Pour les systèmes d'ordre 2, seule l'allure de la réponse en fonction de la valeur du facteur d'amortissement est à connaître.</i></p>		

Déterminer la réponse fréquentielle.	Diagramme asymptotique de Bode, allure du diagramme de Bode réel.	S3
<p><i>Commentaire</i></p> <p><i>Les diagrammes de Bode réels sont déterminés à l'aide d'outils numériques ou fournis.</i></p>		
Déterminer les performances d'un système asservi.	<p>Stabilité d'un système asservi :</p> <ul style="list-style-type: none"> - définition ; - dépassement ; - position des pôles dans le plan complexe ; - marges de stabilité. <p>Rapidité d'un système :</p> <ul style="list-style-type: none"> - temps de réponse à 5 % ; - bande passante. <p>Précision d'un système asservi :</p> <ul style="list-style-type: none"> - théorème de la valeur finale ; - écart/erreur statique ; - écart/erreur de traînage ; - sensibilité aux perturbations en régime permanent ; - lien entre la classe de la fonction de transfert en boucle ouverte et l'écart statique ; - lien entre la position de l'intégrateur et la sensibilité aux perturbations. 	S3
<p><i>Commentaires</i></p> <p><i>Seul le diagramme de Bode est au programme.</i></p> <p><i>Pour les systèmes d'ordre 2, les abaques et relations nécessaires sont fournies.</i></p>		
Mettre en œuvre une démarche de réglage d'un correcteur.	Correcteurs proportionnel, proportionnel intégral et à avance de phase.	S3
<p><i>Commentaire</i></p> <p><i>Pour les correcteurs proportionnel intégral et à avance de phase, la démarche de réglage est fournie.</i></p>		
Caractériser le mouvement d'un repère par rapport à un autre repère.	<p>Trajectoire d'un point.</p> <p>Mouvements de translation, de rotation, ou composé.</p> <p>Vecteurs vitesse et accélération d'un point.</p> <p>Torseur cinématique.</p>	S1
<p><i>Commentaire</i></p> <p><i>Les méthodes graphiques peuvent être présentées mais leur maîtrise n'est pas exigée.</i></p>		

Déterminer les relations entre les grandeurs géométriques ou cinématiques.	Loi entrée-sortie géométrique. Loi entrée-sortie cinématique. Transmetteurs de puissance (vis-écrou, roue et vis sans fin, trains d'engrenages simples et épicycloïdaux, pignon-crémaillère et poulies-courroie).	S1
Déterminer les actions mécaniques en statique.	Référentiel galiléen. Principe Fondamental de la Statique. Principe des actions réciproques.	S2
<p><i>Commentaire</i></p> <p><i>Les méthodes graphiques peuvent être présentées mais leur maîtrise n'est pas exigée.</i></p>		
Déterminer les actions mécaniques en dynamique dans le cas où le mouvement est imposé.	Torseurs cinétique et dynamique d'un solide par rapport à un référentiel galiléen. Principe Fondamental de la Dynamique. Énergie cinétique d'un solide ou ensemble de solides en mouvement par rapport à un référentiel galiléen. Inertie et masse équivalentes.	S3
Déterminer la loi de mouvement dans le cas où les efforts extérieurs sont connus.	Puissance d'une action mécanique extérieure à un solide ou à un ensemble de solides, dans son mouvement par rapport au repère galiléen. Puissance des inter-efforts dans un ensemble de solides. Théorème de l'énergie cinétique. Rendement en régime permanent.	
<p><i>Commentaire</i></p> <p><i>La détermination du moment dynamique peut être introduite sur quelques cas simples mais sa maîtrise ne sera ni exigible ni évaluée. La détermination des actions mécaniques inconnues peut être menée par l'usage d'un outil de simulation numérique.</i></p>		
Déterminer les grandeurs relatives au comportement d'une poutre.	Torseur de cohésion. Sollicitations (traction-compression, torsion et flexion). Contraintes dans une section droite. Déplacements le long d'une ligne moyenne, conditions aux limites. Coefficient de sécurité et résistance mécanique.	S4
<p><i>Commentaire</i></p> <p><i>Les sollicitations ne seront pas combinées. L'effet du cisaillement est hors programme.</i></p>		

Déterminer les signaux électriques dans les circuits.	<p>Circuits en régime alternatif sinusoïdal.</p> <p>Diagramme de Fresnel.</p> <p>Puissance active (continu, monophasé et triphasé en régime alternatif sinusoïdal).</p> <p>Puissances apparente, réactive et facteur de puissance, en monophasé et triphasé en régime alternatif sinusoïdal.</p> <p>Ondulation des grandeurs électriques en régime permanent dans les modulateurs.</p> <p>Composants de filtrage.</p>	S3
Déterminer les conditions d'équilibre de l'association convertisseur électromécanique et charge.	<p>Caractéristiques mécaniques actionneur-charge.</p> <p>Point de fonctionnement.</p>	S4
Caractériser le point de fonctionnement en régime permanent de l'association convertisseur électromécanique et charge.	<p>Stabilité d'un point de fonctionnement.</p> <p>Quadrants de fonctionnement.</p>	

C3 – Mettre en œuvre une démarche de résolution numérique

Compétences développées	Connaissances associées	Semestre
Mener une simulation numérique. $\simeq I$	<p>Choix des grandeurs physiques.</p> <p>Choix des paramètres du solveur (pas de la discrétisation et durée de la simulation).</p> <p>Influence des paramètres du modèle sur les performances.</p>	S4
<p><i>Commentaire</i></p> <p><i>L'influence du choix du solveur (méthode à pas fixe ou à pas variable) pourra être analysée mais aucune connaissance spécifique n'est exigible.</i></p>		
Résoudre numériquement une équation ou un système d'équations. $\simeq I$	<p>Réécriture des équations.</p> <p>Méthodes de dichotomie et de Newton pour résoudre des problèmes du type $f(x) = 0$.</p> <p>Intégration et dérivation numérique (schémas arrière et avant).</p> <p>Schéma d'Euler explicite pour résoudre une équation ou un système d'équations différentielles.</p>	S3

<p><i>Commentaires</i></p> <p>La « réécriture des équations » signifie :</p> <ul style="list-style-type: none"> – remettre en forme des équations pour leurs traitements par une bibliothèque ; – mettre sous forme matricielle un problème (problème de Cauchy et système linéaire). <p>Les méthodes numériques sont introduites au fur et à mesure, en fonction des besoins de la formation. L'utilisation des bibliothèques préimplémentées est privilégiée.</p> <p>Les aspects théoriques liés aux méthodes numériques ne sont pas exigibles (stabilité, convergence, conditionnement de matrices...).</p>		
<p>Résoudre un problème en utilisant une solution d'intelligence artificielle. $\hookrightarrow I$</p>	<p>Apprentissage supervisé.</p> <p>Phases d'apprentissage et d'inférence.</p> <p>Mise en œuvre des algorithmes (k plus proches voisins et régression linéaire monovariée).</p> <p>Choix des données d'apprentissage.</p> <p>Choix des paramètres de classification.</p>	<p>S3</p>
<p><i>Commentaires</i></p> <p>L'apprentissage non supervisé est évoqué, mais aucune connaissance spécifique n'est exigée.</p> <p>L'utilisation des bibliothèques préimplémentées est privilégiée.</p>		
<p>Effectuer des traitements à partir de données de mesures expérimentales. $\hookrightarrow I$</p>	<p>Traitement de fichiers de données.</p> <p>Moyenne et écart type.</p> <p>Moyenne glissante et filtres numériques passe-bas du premier et du second ordre.</p>	<p>S2</p>

D – Expérimenter

D1 – Découvrir le fonctionnement d'un système pluritechnologique et le mettre en œuvre

Compétences développées	Connaissances associées	Semestre
<p>Mettre en œuvre un système en suivant un protocole dans le respect des règles de sécurité.</p>	<p>Normes de sécurité.</p>	<p>S1</p>
<p><i>Commentaires</i></p> <p>Le respect des instructions de sécurité et des normes est une exigence qui s'appuie sur la fourniture des extraits de textes réglementaires.</p> <p>La connaissance des normes de sécurité n'est pas exigible.</p>		
<p>Identifier les constituants réalisant les principales fonctions des chaînes d'information et de puissance.</p>	<p>Chaîne d'information.</p> <p>Chaîne de puissance.</p>	<p>S1</p>
<p>Identifier les principales grandeurs physiques d'effort et de flux.</p>	<p>Grandeurs d'effort et de flux.</p>	<p>S2</p>

D2 – Proposer et justifier un protocole expérimental

Compétences développées	Connaissances associées	Semestre
Proposer un protocole en fonction de l'objectif visé.		S4
Configurer et régler le système en fonction de l'objectif visé.		S2
Choisir la grandeur physique à mesurer ou justifier son choix.	Caractérisation des grandeurs physiques (unité, ordre de grandeur, amplitude, fréquence, valeurs efficace et moyenne, spectre). Appareils de mesure. Capteurs.	S3
Justifier le choix d'un appareil de mesure ou d'un capteur vis-à-vis de la grandeur physique à mesurer.		
Choisir les grandeurs d'entrées à imposer et les grandeurs de sorties à acquérir pour identifier un modèle de comportement sur un système ou sur un constituant du système.	Réponses temporelles et harmoniques.	S2
<p><i>Commentaire</i></p> <p><i>L'objectif consiste à qualifier et quantifier une performance et/ou renseigner un modèle de comportement.</i></p>		

D3 – Mettre en œuvre un protocole expérimental

Compétences développées	Connaissances associées	Semestre
Mettre en œuvre un appareil de mesure adapté à la caractéristique de la grandeur à mesurer.	Règles de raccordement des appareils de mesure et des capteurs. Caractéristiques (calibre, position, etc.) et fonctions d'un appareil de mesure.	S1
Identifier les erreurs de mesure et de méthode.	Incertitudes, résolution, justesse, fidélité, linéarité et sensibilité. Échantillonnage, repliement de spectre, quantification.	S2
<p><i>Commentaire</i></p> <p><i>L'incertitude renvoie à la technologie des appareils de mesure et des capteurs. Il n'est pas exigé de longs développements théoriques et calculs associés.</i></p>		
Générer un programme et l'implanter dans le système cible.	Outils de programmation.	S3
Relever les grandeurs caractéristiques d'un protocole de communication.	Protocole, trame. Débit maximal, débit utile. Caractéristiques des signaux.	S2

Mettre en œuvre une liaison entre objets communicants.	Paramètres de configuration d'un réseau. Internet des objets. Adressage physique et logique.	S4
--	--	----

E – Communiquer

E1 – Rechercher et traiter des informations

Compétences développées	Connaissances associées	Semestre
Rechercher des informations.	Outils de recherche. Mots-clefs.	S2
Distinguer les différents types de documents et de données en fonction de leurs usages.		
Vérifier la pertinence des informations (obtention, véracité, fiabilité et précision de l'information).		
Extraire les informations utiles d'un dossier technique.		
Lire et interpréter un document technique.		S4
<p><i>Commentaires</i></p> <p><i>La veille technologique sera intégrée dans la démarche de recherche documentaire.</i></p> <p><i>Toutes les représentations schématiques technologiques sont fournies, et en particulier les symboles des constituants pneumatiques, hydrauliques et thermiques.</i></p> <p><i>La description d'un même système doit utiliser plusieurs modes de représentations.</i></p>		
Classer les informations.		S2
Effectuer une synthèse des informations disponibles dans un dossier technique.		

E2 – Produire et échanger de l'information

Compétences développées	Connaissances associées	Semestre
Choisir un outil de communication adapté au contexte.	Outils multimédias, outils bureautiques.	S2
Travailler de manière collaborative.	Espaces partagés et de stockage, Espace Numérique de Travail (E.N.T).	

<p>Présenter et formaliser une idée.</p> <p>Faire preuve d'écoute et confronter des points de vue.</p> <p>Présenter un protocole, une démarche, une solution en réponse à un besoin.</p> <p>Présenter de manière argumentée une synthèse des résultats.</p>	<p>Tableau, graphique, diaporama, média, carte mentale, schéma, croquis.</p> <p>Placement de la voix, qualité de l'expression, gestion du temps.</p>	S2
<p>Produire des documents techniques adaptés à l'objectif de la communication. $\Leftrightarrow I$</p>	<p>Diagrammes SysML.</p> <p>Chaîne fonctionnelle.</p> <p>Schéma-blocs.</p> <p>Schémas cinématique, électrique, pneumatique, thermique, hydraulique.</p> <p>Graphe de structure.</p> <p>Croquis, représentations 3D et 2D.</p> <p>Spécifications d'algorithmes.</p>	S4
<p><i>Commentaires</i></p> <p><i>La communication est liée à l'ensemble des activités.</i></p> <p><i>Les croquis doivent traduire les intentions de conception, la représentation normalisée ne peut être exigée.</i></p> <p><i>La description d'un même système doit utiliser plusieurs modes de représentations.</i></p>		
<p>Utiliser un vocabulaire technique, des symboles et des unités adéquats.</p>	<p>Grandeurs utilisées :</p> <ul style="list-style-type: none"> - unités du système international ; - homogénéité des grandeurs. 	S4
<p><i>Commentaire</i></p> <p><i>Les résultats importants et les éléments de synthèse, dans une publication ou une communication orale, sont mis en valeur.</i></p>		

F – Concevoir

F1 – Écoconcevoir l'architecture d'un système innovant

Compétences développées	Connaissances associées	Semestre
<p>Proposer une architecture fonctionnelle et structurelle.</p>		S4
<p><i>Commentaires</i></p> <p><i>Cette proposition peut se faire sous forme d'association de blocs.</i></p> <p><i>Il s'agit d'allouer des constituants à la satisfaction d'exigences fonctionnelles et de décrire les interfaces entre ces constituants.</i></p> <p><i>L'activité de projet est une modalité pédagogique à privilégier pour développer cette compétence.</i></p>		

Intégrer les contraintes d'écoconception dans les architectures proposées.	Cycle de vie, stratégies d'écoconception.	S4
<p><i>Commentaires</i></p> <p><i>Les axes suivants sont abordés :</i></p> <ul style="list-style-type: none"> – développement d'une solution innovante ; – sélection des matériaux ; – réduction de la quantité de matière ; – optimisation des techniques de production ; – optimisation de la logistique ; – réduction de l'impact environnemental de la phase d'utilisation ; – optimisation de la durée de vie du produit ; – gestion de la fin de vie du système. <p><i>Aucune méthode normalisée ne donne lieu à évaluation.</i></p>		

F2 – Proposer et choisir des solutions techniques

Compétences développées	Connaissances associées	Semestre
Proposer et hiérarchiser des critères de choix.	Critères liés au triptyque Matière, Énergie, Information.	S4
<p><i>Commentaire</i></p> <p><i>Les critères de choix abordés sont liés aux fonctionnalités techniques, au respect des normes écologiques et de sécurité, au coût et à l'impact environnemental.</i></p>		
Choisir la technologie des constituants de la chaîne d'information.	Technologie de capteur. Carte de commande.	S4
Choisir la technologie des constituants de la chaîne de puissance.	Dispositifs de stockage d'énergie. Modulateurs de puissance. Convertisseurs de puissance. Transmetteurs de puissance.	S4
<p><i>Commentaires</i></p> <p><i>Pour les modulateurs et convertisseurs de puissance, on se limitera aux technologies électrique, hydraulique, mécanique, pneumatique et thermique.</i></p> <p><i>Les choix seront faits à partir de documents techniques fournis en prenant en compte l'impact environnemental.</i></p>		
Modifier la commande pour faire évoluer le comportement du système.	Modification d'un programme : – système séquentiel ; – structures algorithmiques. Choix du type de correcteur.	S4

F3 – Dimensionner une solution technique choisie dans une démarche de développement durable

Compétences développées	Connaissances associées	Semestre
Dimensionner un constituant des chaînes fonctionnelles.	Constituants de la chaîne d'information. Constituants de la chaîne de puissance.	S4
<p><i>Commentaires</i></p> <p><i>Le dimensionnement peut se faire à partir des exigences et d'une documentation technique tout en respectant les enjeux environnementaux.</i></p> <p><i>Pour la chaîne d'information, le dimensionnement concerne les constituants suivants :</i></p> <ul style="list-style-type: none"> – capteurs (étendue de mesure, précision, justesse, fidélité et sensibilité) ; – CNA – CAN (fréquence d'échantillonnage, résolution et plage de conversion) ; – carte de commande (résolution et mémoire) ; – filtre passif (fréquence de coupure, gabarit) ; – liaisons série et réseaux (débit de communication). <p><i>Pour la chaîne de puissance, le dimensionnement concerne les constituants suivants :</i></p> <ul style="list-style-type: none"> – dispositifs de stockage d'énergie ; – modulateurs de puissance : grandeurs flux et effort en régime permanent et contrainte maximale ; – convertisseurs de puissance : couple quadratique (thermique) équivalent ; – éléments des transmetteurs de puissance modélisés par des poutres : flèche et coefficient de sécurité vis-à-vis de la contrainte maximale ; – organes de guidage dans les transmissions, les formules constructeur sont fournies. 		
Écoconcevoir une pièce en optimisant le triptyque produit-procédés-matériaux.	Familles et propriétés des matériaux. Méthode de choix des matériaux et des procédés. Indicateur de performances. Résistance et déplacement. Influence du procédé sur la géométrie du produit.	S4
<p><i>Commentaire</i></p> <p><i>L'utilisation de logiciels de simulation et de choix de matériaux et procédés est à privilégier.</i></p>		

G – Réaliser

G1 – Réaliser tout ou partie d'un prototype

Compétences développées	Connaissances associées	Semestre
Réaliser tout ou partie de la chaîne de puissance.		S4
Instrumenter tout ou partie d'un système pour valider une exigence ou renseigner un modèle de comportement.		
Intégrer les constituants correspondant à une fonction dans un prototype.		
Implémenter et exécuter un programme sur une cible.		
Valider le fonctionnement du prototype.		
<p><i>Commentaires</i></p> <p><i>La réalisation d'une chaîne d'asservissement numérique pourra servir de support.</i></p> <p><i>L'acquisition de savoir-faire professionnels est exclue.</i></p> <p><i>Tous les moyens de réalisation ou de prototypage rapide mutualisés dans l'établissement scolaire peuvent être utilisés, en particulier les ressources disponibles dans un espace d'innovation partagé (fablab).</i></p>		



Classes préparatoires aux grandes écoles

Filière scientifique

Voie Technologie et sciences industrielles (TSI)

Annexe 4

Programmes d'informatique 1^{ère} et 2^{nde} années

Table des matières

1	Programme du premier semestre	5
2	Programme du second semestre	6
2.1	Méthodes de programmation et analyse des algorithmes	6
2.2	Représentation des nombres	6
2.3	Bases des graphes	7
3	Programme du troisième semestre	8
3.1	Bases de données	8
3.2	Dictionnaires et algorithmes pour l'intelligence artificielle	9
3.3	Algorithmique numérique	9
A	Langage Python	10

Introduction au programme

Les objectifs du programme Le programme d'informatique de TSI et TPC 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 ingénieurs et ingénieures, 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 ;
- que les concepts à enseigner sont les mêmes dans toutes les filières mais que le professeur d'informatique de chaque classe peut adapter la façon de les transmettre et les exemples concrets sur lesquels il s'appuie au profil de ses élèves et aux autres enseignements qu'ils suivent.

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, que ce soit en démontrant un algorithme par une preuve mathématique ou 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 ainsi que ceux auxquels les étudiants sont

familiarisés et qui peuvent être attendus à condition qu'ils soient accompagnés d'une documentation. 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 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 deux premiers 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.

1 Programme du premier semestre

Le programme du premier semestre poursuit 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;
- introduire les premiers éléments de complexité des algorithmes : on ne présente que l'estimation asymptotique du coût dans le cas le pire;
- introduire des outils de validation : variants et invariants.

Le tableau ci-dessous présente les thèmes qui sont abordés lors de ces séances, et, en colonne de droite, une liste, sans aucun caractère impératif, d'exemples d'activités qui peuvent être proposées aux étudiants. 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.
Recherche séquentielle dans un tableau unidimensionnel. Dictionnaire.	Recherche d'un élément. Recherche du maximum, du second maximum. Comptage des éléments d'un tableau à l'aide d'un dictionnaire. <i>Manipulations élémentaires d'un tableau unidimensionnel. Utilisation de dictionnaires en boîte noire. Notions de coût constant, de coût linéaire.</i>
Algorithmes opérant sur une structure séquentielle par boucles simples ou imbriquées.	Recherche d'un facteur (ou d'un mot) dans un texte. Calcul d'une intégrale par la formule de la moyenne ou par la méthode des trapèzes. Tri à bulles. <i>Notion de complexité quadratique. On propose des outils pour valider la correction de l'algorithme.</i>
Utilisation de modules, de bibliothèques.	Lecture d'un fichier de données simples. Calculs statistiques sur ces données. Représentation graphique (histogrammes, etc.).
Algorithmes dichotomiques.	Recherche dichotomique dans un tableau trié. Calcul d'une solution de l'équation $f(x) = 0$ sur $[a, b]$ quand $f(a).f(b) < 0$. <i>On met en évidence une accélération entre complexité linéaire d'un algorithme naïf et complexité logarithmique d'un algorithme dichotomique. On met en œuvre des jeux de tests, des outils de validation.</i>
Fonctions récursives.	Version récursive d'algorithmes dichotomiques. Fonctions produisant à l'aide de <code>print</code> successifs des figures alphanumériques. Dessins de fractales. <i>On évite de se cantonner à des fonctions mathématiques (factorielle, suites récurrentes). On peut montrer le phénomène de dépassement de la taille de la pile.</i>
Tableau de pixels et images.	Algorithmes de rotation de 90 degrés, de réduction ou d'agrandissement. Modification d'une image : flou, détection de contour, etc. <i>Les images servent de support à la présentation de manipulations de tableaux à deux dimensions.</i>
Tris.	Algorithmes quadratiques : tri par insertion, par sélection. Tri par partition-fusion. Tri par comptage. <i>On fait observer différentes caractéristiques (par exemple, stable ou non, en place ou non, comparatif ou non, etc).</i>

2 Programme du second semestre

2.1 Méthodes de programmation et analyse des algorithmes

Même si on ne prouve pas systématiquement tous les algorithmes, on dégage l'idée qu'un algorithme doit se prouver et que sa programmation doit se tester.

Notions	Commentaires
Instruction et expression. Effet de bord.	On peut signaler par exemple que le fait que l'affectation soit une instruction est un choix des concepteurs du langage Python et en expliquer les conséquences.
Spécification des données attendues en entrée, et fournies en sortie/retour.	On entraîne les étudiants à accompagner leurs programmes et leurs fonctions d'une spécification. Les signatures des fonctions sont toujours précisées.
Annotation d'un bloc d'instructions par une précondition, une postcondition, une propriété invariante.	Ces annotations se font à l'aide de commentaires.
Assertion.	L'utilisation d'assertions est encouragée par exemple pour valider des entrées. La levée d'une assertion entraîne l'arrêt du programme. Ni la définition ni le rattrapage des exceptions ne sont au programme.
Explication et justification des choix de conception ou programmation.	Les parties complexes de codes ou d'algorithmes font l'objet de commentaires qui l'éclairent en évitant la paraphrase. Le choix des collections employées (par exemple, liste ou dictionnaire) est un choix éclairé.
Terminaison. Variant. Invariant.	On montre sur plusieurs exemples que la terminaison peut se démontrer à l'aide d'un variant de boucle. Sur plusieurs exemples, on explicite, sans insister sur aucun formalisme, des invariants de boucles en vue de montrer la correction des algorithmes.
Jeu de tests associé à un programme.	Il n'est pas attendu de connaissances sur la génération automatique de jeux de tests; un étudiant doit savoir écrire un jeu de tests à la main, donnant à la fois des entrées et les sorties correspondantes attendues. On sensibilise, par des exemples, à la notion de partitionnement des domaines d'entrée et au test des limites.
Complexité.	On aborde la notion de complexité temporelle dans le pire cas en ordre de grandeur. On peut, sur des exemples, aborder la notion de complexité en espace.

2.2 Représentation des nombres

On présente sans formalisation théorique les enjeux de la représentation en mémoire des nombres. Ces notions permettent d'expliquer certaines difficultés rencontrées et précautions à prendre lors de la programmation ou de l'utilisation d'algorithmes de calcul numérique dans les disciplines qui y recourent.

Notions	Commentaires
Représentation des entiers positifs sur des mots de taille fixe.	La conversion d'une base à une autre n'est pas un objectif de formation.
Représentation des entiers signés sur des mots de taille fixe.	Complément à deux.
Entiers multi-précision de Python.	On les distingue des entiers de taille fixe sans détailler leur implémentation. On signale la difficulté à évaluer la complexité des opérations arithmétiques sur ces entiers.
Distinction entre nombres réels, décimaux et flottants.	On montre sur des exemples l'impossibilité de représenter certains nombres réels ou décimaux dans un mot machine

Représentation des flottants sur des mots de taille fixe. Notion de mantisse, d'exposant.	On signale la représentation de 0 mais on n'évoque pas les nombres dénormalisés, les infinis ni les NaN. Aucune connaissance liée à la norme IEEE-754 n'est au programme.
Précision des calculs en flottants.	On insiste sur les limites de précision dans le calcul avec des flottants, en particulier pour les comparaisons. On n'entre pas dans un comparatif des différents modes d'arrondi.

2.3 Bases des 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	Commentaires
Vocabulaire des graphes.	Graphe orienté, graphe non orienté. Sommet (ou nœud); arc, arête. Boucle. Degré (entrant et sortant). Chemin d'un sommet à un autre. Cycle. Connexité dans les graphes non orientés. On présente l'implémentation des graphes à l'aide de listes d'adjacence (rassemblées par exemple dans une liste ou dans un dictionnaire) et de matrice d'adjacence ainsi que les différences en terme d'occupation mémoire. On n'évoque ni multi-arcs ni multi-arêtes.
Notations.	Graphe $G = (S, A)$, degrés $d(s)$ (pour un graphe non orienté), $d_+(s)$ et $d_-(s)$ (pour un graphe orienté).
Pondération d'un graphe. Étiquettes des arcs ou des arêtes d'un graphe.	On motive l'ajout d'information à un graphe par des exemples concrets.
Fonctions de manipulation.	Obtention du nombre de sommets, ajout/suppression/test d'existence d'un arc ou d'une arête, construction de la liste des voisins d'un sommet, etc. On présente l'importance de construire des fonctions de manipulation élémentaires en vue d'une programmation modulaire ainsi que l'impact du choix d'une représentation en terme de complexité temporelle.
Parcours d'un graphe.	On introduit à cette occasion les piles et les files; on souligne les problèmes d'efficacité posés par l'implémentation des files par les listes de Python et l'avantage d'utiliser un module dédié tel que <code>collections.deque</code> . Détection de la présence de cycles ou de la connexité d'un graphe non orienté.

3 Programme du troisième semestre

3.1 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. On ne présente pas l'algèbre relationnelle ni le calcul relationnel.

Notions	Commentaires
Vocabulaire des bases de données : tables ou relations, attributs ou colonnes, domaine, schéma de tables, enregistrements ou lignes, types de données.	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.
Clef primaire.	Une clef primaire n'est pas forcément associée à un unique attribut même si c'est le cas le plus fréquent. La notion d'index est hors programme. On présente la notion de clef étrangère.
Requêtes SELECT avec simple clause WHERE (sélection), projection, renommage AS. Utilisation des mots-clefs DISTINCT, LIMIT, OFFSET, ORDER BY.	Les opérateurs au programme sont +, -, *, / (on passe outre les subtilités liées à la division entière ou flottante), =, <>, <, <=, >, >=, AND, OR, NOT.
Opérateurs ensemblistes UNION, INTERSECT et EXCEPT, produit cartésien.	
Jointures internes T_1 JOIN T_2 ... JOIN T_n ON ϕ . Autojointure.	On présente les jointures en lien avec la notion de relations entre tables. On se limite aux équi-jointures : ϕ est une conjonction d'égalités. On fera le lien avec la notion de clef étrangère
Agrégation avec les fonctions MIN, MAX, SUM, AVG et COUNT, y compris avec GROUP BY.	Pour la mise en œuvre des agrégats, on s'en tient à la norme SQL99. On présente quelques exemples de requêtes imbriquées. On marque la différence entre WHERE et HAVING sur des exemples.
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. Récupérer le résultat d'une requête à partir d'un programme n'est pas un objectif. Même si aucun formalisme graphique précis n'est au programme, on peut décrire les entités et les associations qui les lient au travers de diagrammes sagittaux informels. Sont hors programme : la notion de modèle logique <i>vs</i> 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.	

3.2 Dictionnaires et algorithmes pour l'intelligence artificielle

Les dictionnaires sont utilisés en boîte noire dès la première année; les principes de leur fonctionnement sont présentés en deuxième année.

Cette partie permet aussi de revisiter les notions de programmation et de représentation de données par un graphe, qui sont vues en première année, en les appliquant à des enjeux contemporains.

Notions	Commentaires
Dictionnaires, clefs et valeurs.	On présente les principes du hachage, et les limitations qui en découlent sur le domaine des clefs utilisables.
Usage des dictionnaires en programmation Python.	Syntaxe pour l'écriture des dictionnaires. Parcours d'un dictionnaire.
Algorithme des k plus proches voisins avec distance euclidienne.	Matrice de confusion. Lien avec l'apprentissage supervisé.
Algorithme des k -moyennes.	Lien avec l'apprentissage non-supervisé. La démonstration de la convergence n'est pas au programme. On observe des convergences vers des minima locaux.
Jeux d'accessibilité à deux joueurs sur un graphe. Stratégie. Stratégie gagnante. Position gagnante.	On considère des jeux à deux joueurs (J_1 et J_2) modélisés par des graphes bipartis (l'ensemble des états contrôlés par J_1 et l'ensemble des états contrôlés par J_2). Il y a trois types d'états finals : les états gagnants pour J_1 , les états gagnants pour J_2 et les états de match nul. On ne considère que les stratégies sans mémoire.
Notion d'heuristique.	On présente la notion d'heuristique à partir d'exemples simples comme le problème du sac à dos.

Mise en œuvre

La connaissance dans le détail des algorithmes de cette section n'est pas un attendu du programme. Les étudiants acquièrent une familiarité avec les idées sous-jacentes qu'ils peuvent réinvestir dans des situations où les modélisations et les recommandations d'implémentation sont guidées, notamment dans leurs aspects arborescents.

3.3 Algorithmique numérique

Dans cette partie du programme, on présente des algorithmes numériques pour une approche pluri-disciplinaire.

On se concentre sur la bonne compréhension de leur fonctionnement et la mise en avant de leur limites.

Ces algorithmes seront appliqués à des problématiques concrètes étudiées dans d'autres disciplines. On veillera à faire programmer par les étudiants les algorithmes étudiés.

Notions	Commentaires
Méthode d'Euler	Résolution approchée d'équations différentielles ordinaires d'ordre 1 et 2. Illustration de l'impact du pas sur la qualité de la solution obtenue.
Méthode de Gauss avec recherche partielle du pivot.	Résolution des systèmes linéaires inversibles. Erreurs d'arrondis et problème de la comparaison à zéro. On aura recours à une conception modulaire pour présenter cet algorithme.
Interpolation polynomiale de Lagrange.	Interpolation par morceaux.

Mise en œuvre

La connaissance dans le détail des algorithmes de cette section n'est toujours pas un attendu du programme.

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

- Typage dynamique : l'interpréteur détermine le type à la volée lors de l'exécution du code.
- 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.
- Appel de fonction par valeur : l'exécution de $f(x)$ évalue d'abord x puis exécute f avec la valeur calculée.

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` (et leur caractère paresseux).
- Comparaisons `==`, `!=`, `<`, `>`, `<=`, `>=`.

Types structurés

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

Structures de contrôle

- Instruction d'affectation avec `=`. Dépaquetage de tuples.
- Instruction conditionnelle : `if`, `elif`, `else`.
- Boucle `while` (sans `else`). `break`, `return` dans un corps de boucle.
- Boucle `for` (sans `else`) et itération sur `range(a, b)`, une chaîne, un tuple, 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, ...`
- Manipulation de fichiers texte (la documentation utile de ces fonctions doit être rappelée; tout problème relatif aux encodages est éludé) : `open`, `read`, `readline`, `readlines`, `split`, `write`, `close`.
- Assertion : `assert` (sans message d'erreur).