

Capacités numériques au programme (PCSI/PC).

4 novembre 2023

1 Capacités présentes en PCSI et en PC :

- **Incertitudes-types composées** : simuler, à l'aide d'un langage de programmation ou d'un tableur, un processus aléatoire permettant de caractériser la variabilité de la valeur d'une grandeur composée.
- **Régression linéaire** : simuler, à l'aide d'un langage de programmation ou d'un tableur, un processus aléatoire de variation des valeurs expérimentales de l'une des grandeurs – simulation Monte-Carlo – pour évaluer l'incertitude sur les paramètres du modèle.

2 Capacités présentes en PCSI.

- **Circuit linéaire du premier ordre** : mettre en œuvre la méthode d'Euler à l'aide d'un langage de programmation pour simuler la réponse d'un système linéaire du premier ordre à une excitation de forme quelconque.
- **Filtrage linéaire** : simuler, à l'aide d'un langage de programmation, l'action d'un filtre sur un signal périodique dont le spectre est fourni. Mettre en évidence l'influence des caractéristiques du filtre sur l'opération de filtrage.
- **Petits mouvements au voisinage d'une position d'équilibre stable, approximation locale par un puits de potentiel harmonique** : à l'aide d'un langage de programmation, résoudre numériquement une équation différentielle du deuxième ordre non-linéaire et faire apparaître l'effet des termes non-linéaires.
- **Point matériel soumis à un champ de force centrale conservatif** : à l'aide d'un langage de programmation, obtenir des trajectoires d'un point matériel soumis à un champ de force centrale conservatif.
- **Pendule pesant** : à l'aide d'un langage de programmation, mettre en évidence le non isochronisme des oscillations.
- **Statique dans le champ de pesanteur uniforme** : à l'aide d'un langage de programmation, étudier les variations de température et de pression dans l'atmosphère.

3 Capacités présentes en PC.

- **Approche microscopique du phénomène de diffusion** : à l'aide d'un langage de programmation, simuler la marche au hasard d'un grand nombre de particules à partir d'un centre et caractériser l'étalement spatial de cet ensemble de particules au cours du temps.
- **Équation de la diffusion thermique** : à l'aide d'un langage de programmation, résoudre l'équation de la diffusion thermique à une dimension par une méthode des différences finies dérivée de la méthode d'Euler explicite de résolution des équations différentielles ordinaires.
- **Dynamique dans un référentiel non galiléen** : à l'aide d'un langage de programmation, illustrer un effet lié au caractère non galiléen du référentiel terrestre.
- **Propagation d'un paquet d'ondes dans un milieu non absorbant et faiblement dispersif : vitesse de phase et vitesse de groupe** : à l'aide d'un langage de programmation, simuler la propagation d'un paquet d'ondes dans un milieu dispersif et visualiser le phénomène d'étalement.

4 Outils numériques : programme officiel.

4.1 PCSI.

Domaines numériques	Capacités exigibles
1. Outils graphiques	
Représentation graphique d'un nuage de points.	Utiliser les fonctions de base de la bibliothèque matplotlib pour représenter un nuage de points.
Représentation graphique d'une fonction.	Utiliser les fonctions de base de la bibliothèque matplotlib pour tracer la courbe représentative d'une fonction.
Courbes planes paramétrées.	Utiliser les fonctions de base de la bibliothèque matplotlib pour tracer une courbe plane paramétrée.
2. Équations algébriques	
Résolution d'une équation algébrique ou d'une équation transcendante : méthode dichotomique.	Déterminer, en s'appuyant sur une représentation graphique, un intervalle adapté à la recherche numérique d'une racine par une méthode dichotomique. Mettre en œuvre une méthode dichotomique afin de résoudre une équation avec une précision donnée. Utiliser la fonction bisect de la bibliothèque scipy.optimize (sa spécification étant fournie).
3. Intégration – Dérivation	
Calcul approché d'une intégrale sur un segment par la méthode des rectangles.	Mettre en œuvre la méthode des rectangles pour calculer une valeur approchée d'une intégrale sur un segment.
Calcul approché du nombre dérivé d'une fonction en un point.	Utiliser un schéma numérique pour déterminer une valeur approchée du nombre dérivé d'une fonction en un point.
4. Équations différentielles	
Équations différentielles d'ordre 1.	Mettre en œuvre la méthode d'Euler explicite afin de résoudre une équation différentielle d'ordre 1.
Équations différentielles d'ordre supérieur ou égal à 2	Transformer une équation différentielle d'ordre n en un système différentiel de n équations d'ordre 1. Utiliser la fonction odeint de la bibliothèque scipy.integrate (sa spécification étant fournie).
5. Probabilité - statistiques	
Variable aléatoire.	Utiliser les fonctions de base des bibliothèques random et/ou numpy (leurs spécifications étant fournies) pour réaliser des tirages d'une variable aléatoire. Utiliser la fonction hist de la bibliothèque matplotlib.pyplot (sa spécification étant fournie)

	pour représenter les résultats d'un ensemble de tirages d'une variable aléatoire. Déterminer la moyenne et l'écart-type d'un ensemble de tirages d'une variable aléatoire.
Régression linéaire.	Utiliser la fonction polyfit de la bibliothèque numpy (sa spécification étant fournie) pour exploiter des données. Utiliser la fonction random.normal de la bibliothèque numpy (sa spécification étant fournie) pour simuler un processus aléatoire.

4.2 PC.

Domaines numériques	Capacités exigibles
1. Tableaux	
Tableaux à une ou deux dimensions.	Choisir une structure de données appropriée à la modélisation d'un problème physique. Réaliser des opérations algébriques simples sur des tableaux. Utiliser les fonctions de base de la bibliothèque numpy (leurs spécifications étant fournies) pour manipuler des tableaux.
2. Équations différentielles et équations aux dérivées partielles	
Équation de diffusion à une dimension.	Mettre en œuvre une méthode des différences finies explicite pour résoudre l'équation de diffusion à une dimension en régime variable.