

# Réponses aux questions

Joseph El-Forzli  
Arthur Renard

## 1. Vecteur

**P1.1** Comment représentez-vous ces vecteurs? Comment sont-ils organisés : quels attributs? quelles méthodes? quels droits d'accès?

— Par des *vector*. L'unique attribut utilisé est un *vector*, les méthodes utilisées sont celles demandées en exercice, les droits d'accès sont publics.

**P1.2** Quel choix avez vous fait pour les opérations entre vecteurs de dimensions différentes?

— Une exception est lancée.

## 2. Modularisation

## 3. Révision des vecteurs

**P4.1** Avez-vous ajouté un constructeur de copie? Pourquoi (justifiez votre choix)?

— Non pour le moment ça n'a pas d'intérêt.

**P4.2** Si l'on souhaitait ajouter un constructeur par coordonnées sphériques (deux angles et une longueur) pour les vecteurs de dimension 3,

(a) que cela impliquerait-il au niveau des attributs de la classe?

—

(b) quelle serait la difficulté majeure (voire l'impossibilité) de sa réalisation en C++? (C'est d'ailleurs pour cela qu'on ne vous demande pas de faire un tel constructeur!)

—

**P4.3** Quels opérateurs avez vous introduits?

— Les opérateurs "operator«", "operator!=" et "operator==" pour afficher les Vecteur et les comparer.

## 4. Matrice 3x3

## 5. Premières Toupies

**P6.1** Comment se situe cette classe par rapport à la classe *Toupie* précédemment définie?

— C'est une sous classe de *Toupie*, en effet *ConeSimple* est avant tout une toupie.

## 6. Intégrateurs

**P7.1** Comment avez vous conçu votre classe *Intégrateur*?

— Nous n'avons pas codé une classe *Intégrateur* mais une classe *Intégrable*. Au lieu de voir un intégrateur comme un moyen de faire évoluer les toupies, nous avons vu les toupies comme des objets "intégrables". Notre classe *intégrable* possède donc deux vecteurs (P et dP), une équation d'évolution virtuelle et différents intégrateurs.

**P7.2** Quelle est la relation entre les classes Intégrateur et IntégrateurEulerCromer?

— Il n'y a donc pas de classe *Intégrateur*, mais *EulerCromer* est une méthode de la classe *Intégrable*.

## 7. Système

**P8.1** En termes de POO, quelle est donc la nature de la méthode *dessine()*?

—

**P8.2** Quelle est la bonne façon de le faire dans un cadre de programmation orientée-objet?

—

**P8.3** A quoi faut-il faire attention pour les classes contenant des pointeurs? Quelles solutions peut-on envisager?

—  
**P8.4** Comment représentez vous la classe *Système* ?  
—

8. **Première simulation (mode texte)**

9. **Graphisme**

10. **Indicateurs (invariants, traces)**

**P11.1** Dans quelle(s) classe(s)/fichier(s) mettez-vous ces méthodes/fonctions ?

- Dans *Toupie*, en effet l'énergie, le moment cinétique et les grandeurs du produit mixte ne sont pas des valeurs propres à certaines toupies, seul leur "côté invariant" l'est. De plus, il pourrait être intéressant d'observer comment se comportent ces valeurs avec des toupies qui les font varier.

11. **Toupies générales**

12. **Autres intégrateurs**

**P13.1** Où cela s'intègre-t-il dans votre projet/conception ? Quels changements cela engendrerait-il (ou pas) ?

- Les intégrateurs s'intègrent dans *Intégrable* comme méthodes supplémentaires. Cela n'implique aucun changement, il suffit de rajouter les intégrateurs dans les options de démarrage du programme.