

Outils Numériques pour l'Ingénieur·e en Physique

2023-2024

6N-076-PHY / ONIP-2

Bloc 4 - Prog. Objet (100%)

Concepts étudiés

[NUM] Classes et objets

Mots clefs

Python; Programmation objet; Classes;
Objets; Méthodes; Attributs

Sessions

- 0 Cours(s) - 1h30
- 0 TD(s) - 1h30
- 6 TD(s) Machine - 2h00
- 0 TP(s) - 4h30

Travail

Par binôme

Institut d'Optique
Graduate School, France
<https://www.institutoptique.fr>

GitHub - Digital Methods
<https://github.com/IOGS-Digital-Methods>

Introduction à la programmation orientée objet

Dans le cadre du module **ONIP-2**, vous serez amenés à réaliser un mini-projet orienté photonique parmi deux sujets au choix :

- Carte d'éclairement de sources lumineuses
- Tracé de rayons dans des systèmes optiques à dioptries

D'un point de vue programmation, vous devrez développer ce projet selon les règles de la **programmation orientée objet**.

Aucune fonction ne devra être utilisée en dehors d'un objet.

Déroulement du module

Ce module se déroule sur **6 séances** :

Séance 1 Découverte de la programmation objet

Séances 2 à 5 Réalisation du mini-projet en binôme

Séance 6 Evaluation du mini-projet en binôme

Livrables attendus

Vous aurez 10 minutes lors de la séance 6 pour présenter l'ensemble de vos résultats et vos analyses.

Pour valider cette session, vous devez **présenter** les **livrables suivants** :

1. **Classes commentées** (selon la norme PEP 257) pour générer des objets (voir les deux sujets proposés pour la liste des classes obligatoires)
2. **Graphiques légendés** incluant toutes les données nécessaires à la bonne compréhension des données présentées
3. **Analyse des figures** obtenues

Les critères d'évaluation et les étapes à suivre sont donnés dans chacun des sujets.

Séance 1 - Programmation orientée objet

Dans cette module, vous serez amenés à développer une application selon les principes de la programmation orientée objet.

Afin de vous familiarisez avec les principes de base, la première séance sera consacrée à **l'étude et la mise en oeuvre d'exemples de la programmation orientée objet** en Python : écriture d'une classe, instanciation d'un objet, interaction entre les objets.

Acquis d'Apprentissage Visés

A travers les exemples proposés, vous serez capables de :

CÔTÉ NUMÉRIQUE

1. **Créer des classes** (sources, plan de projection...)
2. **Définir et documenter les méthodes et attributs** de chaque classe

Ressources

Cette séquence est basée sur le langage Python.

Vous pouvez utiliser l'environnement **Spyder 5** inclus dans *Anaconda 3*.

Des tutoriels Python (et sur les bibliothèques classiques : Numpy, Matplotlib or Scipy) sont disponibles à l'adresse : <http://lense.institutoptique.fr/python/>.

Outils Numériques

Fonctions et bibliothèques conseillées :

- **Numpy** gestion de matrices :
 - **arange**
 - **linspace**
 - **logspace**
- **Matplotlib** affichage de données :
 - **plotly**
 - **figure, plot**
 - **subplot**
 - **legend, title**
 - **xlabel, ylabel**
 - **show**
- **Scipy** fonctions scientifiques :
 - **fftpack** sublibrary
 - **fft, ifft**
 - **fftshift**
 - **fftfreq**

Outils avancés :

- **rcParams** de Matplotlib.pyplot pour l'amélioration de l'affichage de courbes

Etapes

*Les représentations graphiques à produire à chacune des étapes seront accompagnées de **renseignements quantitatifs** comme la valeur moyenne de l'éclairement, son écart-type et son écart Pic à Vallée, absolus et relatifs, sur l'ensemble de la zone représentée (ou sur une sous-partie rectangulaire ou circulaire de celle-ci). Par ailleurs, une **représentation en 3D de la position et de l'orientation des N sources** sera utile.*

Etape 1 Carte d'éclairement pour une source ponctuelle - direction perpendiculaire par rapport au plan éclairé

- Création d'une classe LED_source
- Création d'une classe Carte
- Affichage 2D d'une carte avec une source ponctuelle
- Affichage position des sources sur la carte
- Validation du modèle

Etape 2 Carte d'éclairement pour une source ponctuelle - direction quelconque par rapport au plan éclairé

- Modification de la classe LED_source
- Affichage 2D d'une carte avec une source ponctuelle
- Affichage position des sources sur la carte
- Validation du modèle

Etape 3 Carte d'éclairement pour N sources ponctuelles - direction quelconque par rapport au plan éclairé

- Modification de la classe LED_source
- Affichage 2D d'une carte avec une source ponctuelle
- Affichage position des sources sur la carte
- Validation du modèle

Ouverture A Carte d'éclairement 3D

- Affichage 3D d'une carte avec une source ponctuelle avec position des sources sur la carte

Ouverture B Optimisation d'un éclairage

- Optimisation du nombre et de l'orientation de LEDs pour obtenir un flux lumineux donné sur un plan de travail

Ouverture C Réalisation d'une IHM / PyQt6

- Utilisation de PyQt6 pour l'intégration des précédentes fonctions dans une IHM
- Intégration des graphiques avec pyqtgraph
- Possibilité d'ajouter des sources et d'afficher la contribution indépendante de chacune des sources

Critères d'évaluation

Grille à simplifier (bilan Semestre 5)

- **METHODES NUMERIQUE**
 - **Ecriture Matricielle / Vectorielle**
 - * utilisation des méthodes liées aux vecteurs/matrices (Numpy)
 - * aucune boucle **for** inutile
 - **Organisation en actions élémentaires**
 - * les étapes sont découpées en fonctionnalité plus simple à tester
 - **Description des tests de validation**
 - * chaque fonction a été testée
 - * chaque étape a été validée
 - **Organisation des informations à traiter**
 - * les données sont rangées dans des objets bien identifiés
- **PROGRAMMATION**
 - **Ecriture globale du code et commentaires (PEP 8)**
 - * variables et fonctions respectant les conventions d'écriture standard
 - * commentaires utiles
 - **Utilisation, écriture de fonctions**
 - * paramètres et retours pertinents des fonctions
 - **Documentation des fonctions (PEP257)**
 - * paramètres et retours des fonctions sont documentés
 - **Création de classes et d'objets**
 - * classe contenant des attributs et méthodes pertinents
 - * aucune fonction n'est appelée en dehors d'un objet
- **INGENIEUR.E PHYSIQUE**
 - **Graphiques pertinents et légendés**
 - * graphiques scientifiques (axes, titre...)
 - * axes des graphiques légendés (passage temps/fréquence)
 - **Organisation en actions élémentaires**
 - * les étapes sont découpées en fonctionnalité plus simple à tester
 - **Génération de données pertinentes de tests**
 - * choix de la position des sources pertinent
 - **Analyse des données et validation modèle**
 - * comparaison avec la théorie
 - * analyse pertinente des cartes obtenues
- **AVANCEMENT**
 - Etapes 1 et 2 : x 0.5
 - Etapes 1, 2 et 3 : x 0.7
 - Une des ouvertures : x 1.0

Quelques éléments supplémentaires

Modélisation d'une diode électroluminescente

Les sources (par exemple des LEDs) seront modélisées de manière approchée (valable si l'on n'est pas trop près du composant) comme des **sources ponctuelles**. Ces sources ont un **diagramme de rayonnement** possédant une **symétrie de révolution** autour d'un axe orienté.

L'indicatrice de rayonnement pourra être considérée comme gaussienne, et caractérisée par son intensité visuelle vers l'avant sur l'axe I_0 (en candela) et sa largeur totale à mi-hauteur Δ .

Cette indicatrice peut-être modélisée par l'équation suivante :

$$I(\alpha) = I_0 \cdot \exp(-(4 \cdot \ln(2)) \cdot (\alpha/\Delta)^2)$$

où α est l'angle entre la direction d'émission et l'axe de la source ($\alpha \in [0^\circ, 180^\circ]$).

Positionnement d'une source

Le positionnement de la source dans l'espace sera caractérisé par ses coordonnées (x, y, z) et l'orientation de son axe de symétrie par deux angles (θ et ϕ).

Eclairement / Formule de Bouguer

L'éclairement fourni par une source ponctuelle en un point M de l'espace séparé d'une distance d et d'une inclinaison de ψ par rapport à la direction de la source ponctuelle, est données par la relation photométrique suivante :

$$E = \frac{I \cdot \cos(\psi)}{d^2}$$

L'éclairement produit par N sources (incohérentes) est la somme des éclairements produits par chaque source.