



Préambule à CSC4526/JIN4 Développement C++

Coordinateur : Michel SIMATIC





Plan

1. Introduction
2. **CSC4526/JIN4 : une UV inscrite dans la VAP JIN
(Jeux vidéo /
Interactions et collaborations Numériques)**
3. Zoom sur CSC4526/JIN4
4. Conclusion



Plan



- 1. Introduction**
- 2. CSC4526/JIN4 : une UV inscrite dans la VAP JIN
(Jeux vidéo /
Interactions et collaborations Numériques)**
- 3. Zoom sur CSC4526/JIN4**
- 4. Conclusion**

Introduction : Qui est qui ?

1. **Sur Wooclap <https://app.wooclap.com/CPPTSP>, en 3 fois *un* mot (séparés par des espaces) :**
 1. Que vous évoque JIN ?
 2. Votre attente par rapport à CSC4526/JIN4 (**pas JIN**)
 3. Votre crainte par rapport à CSC4526/JIN4 (**pas JIN**)
2. **Debout, depuis votre place :**
 1. Prénom NOM
 2. Vos 3 mots (envoyés à Wooclap quand vous vous levez)
 3. Où vous voyez-vous dans 5 ans ?
 4. Quels langages maîtrisez-vous bien ?
 5. Freins à CSC4526/JIN4 (par exemple, Cassiopée dont l'avancement est insuffisant, CF2, etc.)
3. **Après cette séance, sondage Moodle sur votre parcours de 2A avant d'arriver en CSC4526 et votre Cassiopée**



Plan

1. Introduction

2. CSC4526/JIN4 : une UV inscrite dans la VAP JIN (Jeux vidéo / Interactions et collaborations Numériques)

3. Zoom sur CSC4526/JIN4

4. Conclusion

Organisation des études JIN

Généralités



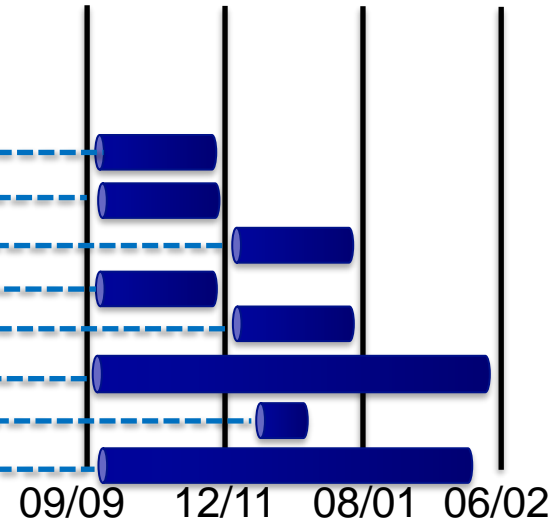
■ S8 spécifique TSP

- CSC4508 : Systèmes d'exploitation
- CSC4526 : Développement C++



■ S9

- Dans les locaux de l'ENSIIE et de TSP
- Ensemble d'UE à la structure ~ filière ENSIIE classique
 - ECOFH2J : SHS, Design & Game Design
 - MER : Modélisation Et Rendu
 - DMIA : [Seulement ENSIIE ? Dév sur Mobile] / IA
 - IEC : Interactions Et Capteurs
 - SIR : Systèmes Interagissant en Réseau
 - PFE: Projet d'approfondissement
 - PROCCI : Profils de Comportement et de Comm.
 - ENV5001 (TSP) :Initiation à la Resp. Sociétale des Entr.



■ S10

- Stage 6 mois en entreprise ou laboratoire (à partir du 10/02/25)
- Géré par l'école d'origine de l'étudiant-e (quid cas étudiant-e-s en mobilité ?)

Planification S9 JIN : Dates phare


- **Proposition de Projet de Fin d'Etude = été 2024**
- **Devoir de vacances = été 2024**
 - On suppose Unity OK
 - Etudiant·e·s TSP : Rattrapage par rapport à VR
 - Etudiant·e·s ENSIIE : Rattrapage par rapport à C++
 - Si possible, vous auto-former à Unreal Engine
 - Révisions sur les produits scalaires (pour cours de rendu)
 - **(Optionnel)** lié à PROCCI, lecture de 1-2 livres
- **Rentrée S9 = 09/09/23 à 9h (TSP)**
 - 09/09 : Après-midi « Intégration »
- **Congés d'automne = 28/10/24 – 01/11/24**
- **Paris Games Week : JIN n'y participe plus**
- **PROCCI = 18/11/24 – 22/11/24**
- **UniJam = 29/11/24 18h – 01/12/24 20h**
- **Congés Noël = 19/12/24 – 05/01/25**
- **Fin des cours JIN = 06/02/25 à 13h**

Oui nide iou !

- **Guillaume Bouyer (co-coordonnateur JIN côté ENSIIE) et Michel SIMATIC sont preneurs d'aide sur l'organisation d'UniJam 2024 (29/11-01/12, dans les locaux du campus TSP)**
 - A quelles étudiant·e·s (1A, 2A) puis-je en parler ?
NB : C'est sans engagement !
 - Déjà des idées de sponsors ?



Plan

1. Introduction
2. **CSC4526/JIN4 : une UV inscrite dans la VAP JIN
(Jeux vidéo /
Interactions et collaborations Numériques)**
-  3. **Zoom sur CSC4526/JIN4**
4. Conclusion

Zoom sur CSC 4526 / JIN4

Acquits d'Apprentissage Visés

- **A l'issue du module, les étudiants sont capables, dans le cadre d'une application interactive (qui peut être un jeu vidéo) de petite taille, mais réaliste de :**
 - spécifier cette application sur la base d'un sujet imposé
 - développer cette application en C++ moderne (utilisation de "Resource Acquisition Is Initialization" [RAII], de pointeurs intelligents), à l'aide de l'environnement de développement (IDE) Visual Studio ou CLion
 - développer des tests unitaires dans le cadre d'une démarche Test-Driven Development (TDD)
 - intégrer 3 bibliothèques externes (dont le moteur graphique SFML ou Qt) à leur application,
 - mettre en œuvre une hiérarchie de classes et expliquer cette hiérarchie de classe à l'aide d'un diagramme UML,
 - intégrer au moins un design pattern (qui peut être un game programming pattern) au sein de l'application,
 - implanter une interface utilisateur intuitive et fluide,
 - présenter et/ou critiquer de manière argumentée, en public, un travail réalisé.

Equipe pédagogique



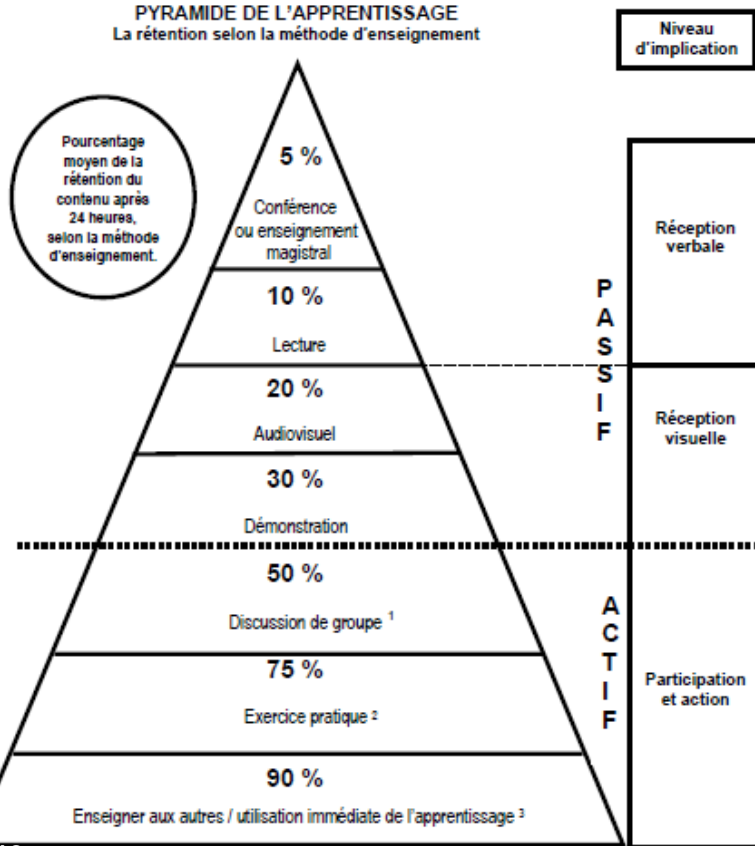
■ **Loïc Joly (SonarSource)**



■ **Michel Simatic (TSP - Coordonnateur)**

Zoom sur CSC4526/JIN4

Grandes lignes



■ Modalités

- Un peu de théorie
- **Beaucoup** de pratique (notamment avec un exercice fil rouge)
- Un micro-projet (jeu ou application interactive, au choix)

■ Pré-requis

- Algorithmique (notions)
- Langage C (bonne pratique)
- Java / Lang. objet (bonne pratique)
- UML (bonne pratique)

Zoom sur CSC4526/JIN4

Déroulement

- **1 séance d'introduction : 06/05**
 - 1 exercice à faire pour le 14/05 8h30
- **9 CI de 3h**
 - Règle des 5 minutes
 - 1 exercice fil rouge à avancer d'une séance à l'autre (faisable en 2023-2024 ?)
 - 1 extrait de chapitre à lire d'une séance à l'autre (faisable en 2023-2024 ?)
- **1 micro-projet du 29/05 au 18/06**
 - 29/05 12h45-13h15 : Sujet
 - 31/05 14h30-16h00 : Présentation des premières idées
 - 01/06 – 18/06 : Soutien sur Discord
 - 20/06 10h00-13h15 & 14h30-17h45 : Soutenances

Zoom sur CSC4526/JIN4

A propos de « *Beaucoup* de pratique »

- « L'important, ce n'est pas la destination, c'est le voyage. » (Robert Louis Stevenson)
- Nous avons besoin de vous pour nous aider à estimer le travail nécessaire pour faire les exercices (et s'imprégner de toutes les notions qu'ils introduisent).
- → Fichier Excel à remplir pour indiquer le temps (en minutes) passé sur chacun des exercices C++
 - <https://partage.imt.fr/index.php/s/p5mgqy8WdixxoMk>
 - → L'an prochain, des statistiques plus précises sur les temps nécessaires
- **Merci d'avance**
 - D'y indiquer ***sincèrement*** les temps passés
 - D'y écrire des « SOS » pour qu'on vienne vous aider
 - Autre manière de dire « SOS » = Changer votre rôle Discord en « EtudiantQuestion »

Zoom sur CSC4526/JIN4

Recommandations

■ Quel état d'esprit adopter vis-à-vis de ce cours ?

- "La phrase la plus excitante à entendre, celle qui annonce de nouvelles découvertes, n'est pas «Eureka», mais plutôt «Tiens, c'est marrant...» " (Isaac Asimov)



■ Suivez ****tous**** les conseils que vous donnent SonarLint et, si vous êtes sous Clion, Clang-Tidy

- Un classique : `static_cast<float>(variable_de_type_double)`



■ Usez et abusez du salon Discord

- en soumettant des questions
- mais aussi en apportant des réponses !

■ Ne dites pas à Loïc ou Michel que vous n'avez jamais vu des pointeurs, de l'UML, des tests unitaires, des patrons de conception, des traitements d'erreur (exception)...

- Loïc et Michel connaissent CSC4102 et CSC4103 😊

Zoom sur CSC4526/JIN4

Annales du micro-projet

■ 2015

- « La vie, c'est comme une bicyclette, il faut avancer pour ne pas perdre l'équilibre. » (Albert Einstein)

■ 2016

- « Si tu veux vraiment l'accomplir, tu trouveras le moyen. Si tu ne veux pas vraiment, tu trouveras l'excuse. » (Jim Rohn)

■ 2017

-



■ 2018

- |00|

■ 2019

- « Entre le fort et le faible, entre le riche et le pauvre, entre le maître et le serviteur, c'est la liberté qui opprime et la loi qui affranchit » (Henri LACORDAIRE)

■ 2020

- « Développez un jeu s'appuyant sur un épisode de la mythologie grecque. »

■ 2021

- « Science sans conscience n'est que ruine de l'âme. » (Rabelais)

■ 2022

- « Développez un jeu inspiré d'un livre de Jules Verne . »

■ 2023

- 3 Monsieur Madame ont été tirés au hasard pour votre binôme/monôme. Votre jeu doit s'inspirer de 2 de ces 3 Monsieur-Madame, en prenant en compte les noms des deux Monsieur-Madame. Dit autrement, cette inspiration peut aussi s'inspirer de l'ensemble du titre des ouvrages (sans toutefois omettre les noms des Monsieur-Madame concerné-e-s).

■ 2024

- ???

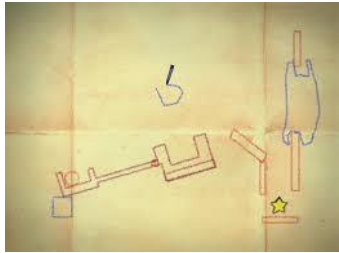
Zoom sur CSC4526/JIN4

Évaluations



■ Industriels/Enseignants-chercheurs → Étudiant(e)s (NB : pas profs / élèves !)

- Micro-projet
 - Beaucoup de temps hors présentiel
 - Occasion de consolider vos connaissances en C++
 - Occasion de vous former à SFML (ou Unreal)
 - Occasion de vous former à ImGui
 - Occasion d'ajouter une ligne à votre CV



■ Étudiant(e)s → Industriels/Enseignants-chercheurs

- Fiche d'évaluation sur Moodle
- Réunion bilan au moment de l'oral de projet

■ Occasion de voir, pour chacun, si la VAP JIN lui correspond (cf. zoom en slide suivant)

Est-ce que JIN est fait pour moi ?

Suis-je fait-e pour JIN ?

■ Abordé en CSC4526

- Ecrire du C++ moderne
- Design patterns
- Games programming patterns
- Tirer le meilleur d'une machine

■ Abordé en CSC4508

- Multi-threading , gestion de la concurrence
- Intérieur d'un système d'exploitation
- Tirer le meilleur d'une machine

Game Engine Architecture, Third Edition (English Edition) 3e Édition, Format Kindle

de Jason Gregory (Auteur) | Format: Format Kindle

★★★★★ 119 évaluations

Format Kindle 60,12 €

Raté 86,19 €

Lisez avec notre Appli gratuite 2 Discours à partir de 100,34 €

11 Neuf à partir de 83,87 €

In this new and improved third edition of the highly popular **Game Engine Architecture**, Jason Gregory draws on his nearly two decades of experience at Midway, Electronic Arts and Naughty Dog to present both the theory and practice of game engine software development. In this book, the broad range of technologies and techniques used by AAA game studios are each explained in detail, and their roles within a real industrial-strength game engine are illustrated.

En lire plus

Langue: Anglais | Format: Print Replica

En raison de la taille importante du fichier, ce livre peut prendre plus de temps à télécharger

Les clients ayant acheté cet article ont également acheté

Image	Titre	Auteur	Format	Évaluations	Prix
	Foundations of Game Engine Development, Volume 1: Mathematics (English Edition)	Eric Lengyel	Format Kindle	★★★★★ 97	36,28 €
	Foundations of Game Engine Development, Volume 2: Rendering (English Edition)	Eric Lengyel	Format Kindle	★★★★★ 50	54,45 €
	Game Programming Patterns	Robert Nyström	Format Kindle	★★★★★ 585	15,89 €
	Design Patterns: Elements of Reusable Object-Oriented Software (English Edition)	Gamma Erich	Format Kindle	★★★★★ 1141	19,19 €
	Real-Time Rendering, Fourth Edition (English Edition)	Tomas Akenine-Möller	Format Kindle	★★★★★ 114	60,12 €
	Game Engine Black Book: Wolfenstein 3D (English Edition)	Fabien Sanglard	Format Kindle	★★★★★ 177	8,50 €

Un sujet que CSC4526/JIN4 ne fait qu'effleurer : La Conception Orientée Data

■ Conception Orientée Data

- La *Conception Orientée Data* est une technique d'optimisation basée sur la mémoire cache des processeurs (CPU). Les données, et les données suivantes adjacentes en RAM sont copiées en mémoire cache CPU pour un accès rapide.
- La *Conception Orientée Data* cherche à tirer parti du mécanisme de *prefetch* de manière optimale. Il s'agit, en général, de stocker les données de plusieurs objets, manipulés les uns à la suite des autres par un certain type d'instruction, de manière contiguë – ou proche, tout du moins – en RAM.

■ Versus Programmation Orientée Objet (POO, enseignée à TSP)

- La POO tend à regrouper des données hétérogènes dans des objets (instances de classes), d'une manière plus simple à appréhender pour le développeur, mais moins efficace pour l'ordinateur.
- Ces objets sont ensuite manipulés par des instructions dont peu utilisent l'ensemble des données (attributs) qu'ils contiennent.
- Les caches CPU sont donc souvent pollués par des données inutiles → Plus d'appels à la RAM (plus de cache misses) que nécessaire, et donc une perte de temps significative.

■ (Merci à Gibril Günder, JIN 2021-2022, et à son Directeur de stage)

CSC4526/JIN4 ne suffit pas !

- **Design Patterns** : <https://refactoring.guru/design-patterns>
 - Game Programming Patterns : <http://gameprogrammingpatterns.com/>
- **Montez en puissance en algorithmie**
 - <https://www.hackerrank.com/>
 - <https://www.codingame.com>
 - <https://projecteuler.net/>
 - Livre « Programmation efficace : les 128 algorithmes qu'il faut avoir compris et codés en Python au cours de sa vie », Christoph Dürr et Jill-Jênn Vie, éditions Ellipses
- **C++**
 - Livre de Stroustrup
 - <https://github.com/isocpp/CppCoreGuidelines>
 - <https://cpp.developpez.com/> (Cours + FAQ)
 - <http://www.cplusplus.com/>
 - <http://en.cppreference.com/w/>
 - <https://www.youtube.com/user/CppCon>
 - <http://www.bfilipek.com/> (blog d'un fan de C++)
 - J'insiste : Suivre les conseils de SonarLint et Clang-Tidy
- **Challenge(s) de code en ligne**
 - Exemple : <http://battledev.blogdumoderateur.com/>

Remarque (avancée) à propos du langage C++

- **C++ est un langage normalisé par un comité non dépendant d'éditeurs de logiciels (<https://isocpp.org/>)**
- **Ce comité normalise une nouvelle version tous les 3 ans.**
 - Version actuelle = C++23
 - Version utilisée en CSC4526 = C++20
 - NB : Souvent, SonarLint vous fera des suggestions d'améliorations liées à C++20 (sur du code récupéré de StackOverflow, ChatGPT, etc.)
 - FYI, depuis [Boost 1.84.0](#) (13/12/2023), "C++03 is no longer supported, the minimum required language version is C++11."
- **Les différents compilateurs n'implémentent pas forcément toutes les fonctionnalités d'une version. En particulier, différences majeures entre Clang et Apple Clang (https://en.cppreference.com/w/cpp/compiler_support/20)**



Plan

1. Introduction
2. **CSC4526/JIN4 : une UV inscrite dans la VAP JIN
(Jeux vidéo /
Interactions et collaborations Numériques)**
3. Zoom sur CSC4526/JIN4
4. Conclusion



Conclusion

- (voix suave) « **Nous vous souhaitons un agréable voyage sur cette UV en notre compagnie** »

- **Questions ?**
 - «A la question "Pourquoi m'est-il si difficile de demander ?", un ange répond : "Parce qu'on t'a étouffé·e quand tu as voulu demander, et tu n'as pas encore tout à fait repris ton souffle." »
(« Dialogues avec l'ange », Gitta MALLASZ)