

Semaine du	CM	TD/TP
02/09	<u>Cours 1</u> Contexte, motivations Concepts prog. objet (survol) exemple C → C++	X = 1H30 (1 séance) XX = 3H (2 séances)
09/09	<u>Cours 2</u> <u>Modélisation objet UML</u> notation norm. classes associations/compo/agreg navigabilité et châtiment classe association (équivalence) associations qualifiées héritage/polym. (bases) exemples, contre exemples	X TD/TP 1 Diagrammes objets Dessiner papier des schémas d'implémentation mémoire données - sémantique valeur vs référence - référence pointeur vs indice vs id → Application fil conducteur : Maillage/Triangle/Point
16/09	<u>Cours 3</u> <u>C++ pratique 1</u> assortiment linguo et concepts les flots cin>> cout<< getline namespace, bool, string, vector (sans les iterateurs...) dimensionnement init. & push... déclaration variables et portée alias (using), auto	XX TD/TP 2 Diagrammes de classe On reprend les exos précédents Variantes. Navigabilité. → Maillage/Triangle/Point suite → Réseau bancaire ... → Parc véhicule d'entreprise Vehicule / VL / Camion / Scoot ... héritage
23/09	<u>Cours 4</u> <u>C++ pratique 2</u> surcharge, enums, new/delete, références, const structs, paramètres par défaut	XX TD/TP 3 Code C++ courant, sans classe perso cin/cout, bool, size_t string, vector, parcours avec .size() C++ : pratiquer le Cours 3
30/09	<u>Cours 5</u> <u>Classes C++ les bases</u> implémenter une classe simple instancier (autom / dynamique) composer (attributs classes)	XX TD/TP 4 Surcharge, references, new/delete sensibilisation au const Utilisation structs simples en C++ C++ : pratiquer le Cours 4
07/10	<u>Cours 6</u> <u>Classes C++ les compléments</u> méthodes définies inline attributs pointeurs attributs dynamiques (cons/des) attributs vector d'objets surcharges opérateurs, bazar : attrib./def., délégation...	XX TD/TP 5 → Une classe simple : Coords - implémenter - utiliser (tester & exploiter) → Une classe composée : Astre - attributs masse et Coords (déclarer / instancier) - implémenter & utiliser

Semaine du	CM	TD/TP
14/10	<u>Cours 7</u> <u>Conteneurs en C++ : la STL</u> Conteneurs, itérateurs, auto algorithmes de la STL vector compléments (1D, 2D ...) set / unordered_set (usager) map / unordered_map (usager)	XX <u>TD/TP 6</u> → Implémentation du modèle Maillage/Triangle/Point <i>suite</i> Mise en œuvre attributs vectors d'objets référencés par pointeurs
28/10	<u>Cours 8</u> <u>Héritage et polymorphisme</u> Héritage simple, hiérarchies, ordres de Construct. Destruct. Héritage multiple (bref) Redéfinitions (overriding) upcasting, slicing Méthode virtuelle (pure ou pas) Polymorphisme, RTTI Classes abstraites (pures ou pas)	X <u>TD/TP 7</u> → Continuation TD/TP6 → Implémentation du modèle Maillage/Triangle/Point <i>suite</i> copie composite complexe utilisation map, set, erase
04/11	<u>Cours 9</u> <u>Abstraction, design patterns</u> Modélisation objet UML <i>suite</i> modèles avancés, patrons de conception (design patterns) quelques exemples (composite..) Granularité des modèles objets	XX <u>TD/TP 8</u> → Exercices d'application avec des « robots » textuels Hiérarchie(s) de classes Méthodes virtuelles Polymorphisme
11/11	<u>Cours 10</u> <u>Exceptions, flots, fichiers</u> Sérialisation Persistance : fichiers Exceptions	XX <u>TD/TP 9</u> → Exercices d'application robots textuels <i>suite</i> pattern stratégie et composite utilisation d'un « framework »
18/11	<u>Cours 11</u> <u>Développement de templates</u> Programmation générique : Templates côté utilisateur Templates côté implémentation	XX <u>TD/TP 10</u> Exercices sur les flots Parsing de chaîne Fichiers Avancé : exceptions, sérialisation
25/11	<u>Cours 12</u> <u>Compléments</u>	XX <u>TD/TP 11</u> Suivi de projet
02/12	<u>Cours 13</u> <u>Synthèses et révisions</u>	XX <u>PROJET</u> Soutenance de projet (projet Ω)