
L'objectif des exercices de cette feuille est de finaliser **la gestion des déplacements des navires** entre les îles, avec gestion des manoeuvres et ... de tous les échouages éventuels !

Exercice 1. Classe *Triangle* – V 2.1.0

Installer avec succès la version fournie en annexe.

Exercice 2. Classe *Triangle* – V 2.1.0 / Prédicats *isocèle*, *equilateral* et *rectangle*

Développer en langage C++ les prédicats précités.

Développer et exécuter avec succès le module de tests unitaires correspondant.

Exercice 3. Classe *Navire* – V 1.2.0

Installer avec succès la version fournie en annexe.

Exercice 4. Classe *Navire* – V 1.2.0 / Méthode *deplacer*

Coder en langage C++ la méthode ***deplacer*** suivant l'algorithme fourni en annexe.

Exécuter avec succès tous les tests unitaires fournis en annexe.

Exercice 5. Classe *Navire* – V 1.2.0 / Méthode *manoeuvrer*

Les changements de direction d'un navire seront modélisés par une consigne d'angle ***alpha*** (radians), fournie par rapport au vecteur vitesse courant. Il en sera de même pour un changement de vitesse (accélération ou ralentissement), avec une consigne ***deltaRho***, fournie sous forme d'un pourcentage.

Les deux consignes seront appliquées au moyen d'une nouvelle méthode ***manoeuvrer*** de la classe ***Navire***.

Un navire ne pourra évidemment pas dépasser sa vitesse maximum, qui est un nouvel attribut de la classe ***Navire***. Cette vitesse limite pourra être modifiée en cours de jeu, en fonction des dégâts sur la coque et/ou les moteurs subis par le navire.

Développer une première version de l'algorithme de la méthode *manoeuvrer*, dans laquelle on ne tiendra pas compte de l'inertie d'un navire, des conditions météo dans la zone de manoeuvre et des dégâts éventuels subis.