

## INFO-H-100 : Introduction à la programmation

### Projet : Touché-Coulé

Le but du projet est d'implémenter en *Python 3* le jeu *Touché-Coulé* (marque déposée)<sup>1</sup>.

Ce jeu se joue à deux personnes. Les deux joueurs positionnent initialement cinq navires sur leur *grille-océan*, dont un porte-avions (longueur 5), un cuirassé (longueur 4), un croiseur (longueur 3), un sous-marin (longueur 3) et un destroyer (longueur 2).

Une fois ce placement effectué, à tour de rôle, chaque joueur annonce les coordonnées d'une cible composée d'une lettre et d'un chiffre (ex : H10) correspondant à un emplacement sur la *grille-océan* adverse.

L'adversaire indique alors si le tir a touché ou manqué la cible. Si la cible a été touchée, il marque le point d'une torpille rouge sur la *grille-océan*.

Le joueur attaquant marque son score sur la *grille-cible* en utilisant une torpille rouge pour une cible touchée et une torpille blanche pour une cible manquée.

Un navire est coulé lorsque celui-ci est touché sur toute sa longueur. Le joueur perdant un navire doit l'annoncer explicitement à son adversaire.

Le premier joueur qui coule tous les navires de la flotte adverse gagne la partie.

## 1 Programme à réaliser

On vous demande de réaliser un jeu *Touché-Coulé*. Votre programme devra se dérouler comme suit :

1. Votre programme chargera en mémoire l'ensemble des navires et leurs longueurs respectives disponible sous la forme d'un fichier texte sur la page web des projets. Ce fichier devra obligatoirement se trouver dans le même répertoire que votre programme. Le format de ce fichier est détaillé ci-dessous.
2. Ensuite, les navires seront placés sur les *grilles-océan* des deux joueurs. Le positionnement des navires du joueur sera réalisé par celui-ci alors que le placement pour l'ordinateur se fera aléatoirement. Deux navires ne peuvent pas se recouvrir.
3. Pour placer un tir, le joueur entre les coordonnées de la case qu'il désire viser. L'ordinateur quant à lui devra soit effectuer un tir aléatoire, soit, si un navire a été préalablement touché mais non coulé, tenter de l'achever.
4. Ensuite, la *grille-cible* est mise à jour avec les informations du tir précédent.
5. La partie se poursuit tant que tous les navires du joueur ou de l'ordinateur n'ont pas été coulés.

Les navires sont contenus dans le fichier `navires.dat`. Chaque ligne se compose de la manière suivante : le nom du type du navire, le nombre de ce type de navire et sa longueur. Extrait du fichier :

```
| croiseur 1 3  
| sous-marin 1 3
```

## 2 Apports personnels

Les apports personnels seront valorisés à hauteur de 5 points sur 20. En d'autres mots, un projet parfait sans apport personnel aura une valeur de 15 points sur 20.

Voici quelques exemples d'apports personnels :

- Permettre l'utilisation de grilles de tailles variables et non juste du 10x10. (simple)
- Permettre l'utilisation de plusieurs navires du même type. (simple)
- Un mode multijoueur. Celui-ci devant être un système tour par tour. (moyenne)
- Affichage amélioré grâce à l'Art ASCII<sup>2</sup>. (moyenne)
- Séparer le code en au moins deux modules (deux fichiers `.py`). Le premier module, le *modèle*, contiendra les fonctions logiques du projet (vérifier qu'un mot est dans le dictionnaire, ajouter un mot au plateau de jeu, etc.). Le second module, la *vue*, contiendra les interactions avec l'utilisateur (c'est-à-dire l'affichage et les entrées) et importera le premier module. C'est donc ce second module qui sera exécuté pour lancer votre jeu *Touché-Coulé*. Attention : pour que la découpe soit bien faite, il faut qu'il n'y ait *aucune* instruction de type `print` ou `input` dans le modèle, et que le modèle n'importe *pas* la vue. La vue doit contenir le moins de "logique" possible. (moyenne)
- Interface graphique. (difficile)

1. Voir [http://regle.jeuxsoc.fr/bnava\\_rg.pdf](http://regle.jeuxsoc.fr/bnava_rg.pdf) pour les règles complètes.

2. [http://fr.wikipedia.org/wiki/Art\\_ASCII](http://fr.wikipedia.org/wiki/Art_ASCII)

# Délivrables

1. Vous devez déposer une version de votre code sur le serveur Git se trouvant à l'adresse <http://wit-projects.ulb.ac.be/rhocode/INFO-H-100/2012/2/>.

**Chaque fonction devra être correctement documentée.** C'est-à-dire pour chaque fonction : son prototype, une description succincte de la tâche qu'elle remplit, ses paramètres et leur domaine (le type et les valeurs possibles), le type de ses valeurs de retour et leur domaine et un exemple d'utilisation. Par exemple<sup>3</sup> :

```
def daysInMonth(month, year):
    """Calcule le nombre de jours d'un mois.

    Arguments:
        month (int) : le mois considere, entre 1 et 12.
        year (int) : l'annee consideree.

    Valeurs de retour:
        int. Retourne le nombre de jours associes au mois et a l'annee considerees.
        Retourne -1 si les parametres ne respectent pas les domaines.

    Exemples:
        >>> print daysInMonth(2, 2012)
        29
        >>> print daysInMonth(13,2012)
        -1
    """
    <<code de la fonction>>
```

Rappelez-vous que la *docstring* est le texte qui sera affiché lorsque vous tapez `help(daysInMonth)` dans l'interpréteur ; il s'agit donc de la documentation nécessaire à un **programmeur** pour *utiliser* la fonction. Ce texte ne s'adresse pas à l'utilisateur du *programme* (le joueur), et n'explique pas le fonctionnement *interne* de la fonction ; il explique uniquement **comment utiliser** la fonction.

## Consignes

Le projet se fera obligatoirement par groupes de deux étudiants de la même série.

1. Le code source sera rendu via le serveur *GIT*, de la même manière que pour le premier projet. On vous demande cette fois de créer votre dépôt dans le groupe INFO-H-100>>2012>>2. Le nom du dépôt devra être <netid1>-<netid2>, où les deux *netids* doivent être dans l'ordre alphabétique.  
Par exemple, gaverhae-mwaumans.
2. Le dépôt *doit* être créé pour le 4 mars 2013 au plus tard et servira à la validation des groupes.
3. Le projet *doit* être remis sur le dépôt Git le 31 mars 2013 à 22h au plus tard. Les défenses seront organisées la semaine du 1<sup>er</sup> avril et celle du 15 avril.

Les critères d'évaluation sont les suivants :

- La résolution du problème énoncé.
- Le respect scrupuleux de l'énoncé.
- Le respect des conventions<sup>4</sup> et des règles de bonne pratique<sup>5</sup> publiées sur le site web des projets.
- Votre défense en salle machine devant les assisants.
- L'efficacité de vos algorithmes et structures de données, en restant bien sûr dans le cadre de la matière vue aux cours et aux séances d'exercices.

Les élèves assistants sont disponibles à 12h30 dans la salle Socrate afin de répondre à vos questions. Consultez leur horaire sur la page web des TPs.

---

3. Basé sur [http://packages.python.org/an\\_example\\_pypi\\_project/sphinx.html#full-code-example](http://packages.python.org/an_example_pypi_project/sphinx.html#full-code-example)

4. [http://cs.ulb.ac.be/public/teaching/infoh100/conventions\\_python](http://cs.ulb.ac.be/public/teaching/infoh100/conventions_python)

5. [http://cs.ulb.ac.be/public/teaching/infoh100/bonne\\_pratique\\_python](http://cs.ulb.ac.be/public/teaching/infoh100/bonne_pratique_python)

# Annexe : Règles officielles

## LE JEU

1. A tour de rôle, chaque joueur annonce les coordonnées d'une cible, composée d'1 lettre et d'1 chiffre. Exemple : H10.
2. L'adversaire indique alors si le tir a touché ou manqué la cible. Si la cible a été touchée, marque le point d'une torpille rouge sur la **grille-océan**.
3. Le joueur attaquant marque son score sur la **grille-cible** en utilisant une torpille rouge pour une cible touchée et une torpille blanche pour une cible manquée.

## COULER UN NAVIRE

Un navire est coulé lorsque tous les trous qui le composent contiennent une torpille rouge. Annonce alors à ton opposant quel navire a été coulé et retire-le de ta grille-océan.

## COULER LA FLOTTE

Le premier joueur qui coule les 5 navires de la flotte adverse gagne la partie !

## RANGEMENT

Fixe les 5 navires n'importe où sur la grille-océan et place les torpilles dans les compartiments de stockage. Ferme les consoles de jeu et fais-les glisser l'une contre l'autre pour qu'elles s'emboîtent.

- JEUX -  
LES BONNS VOYAGES

**MB**  
JEUX

0605 14528101



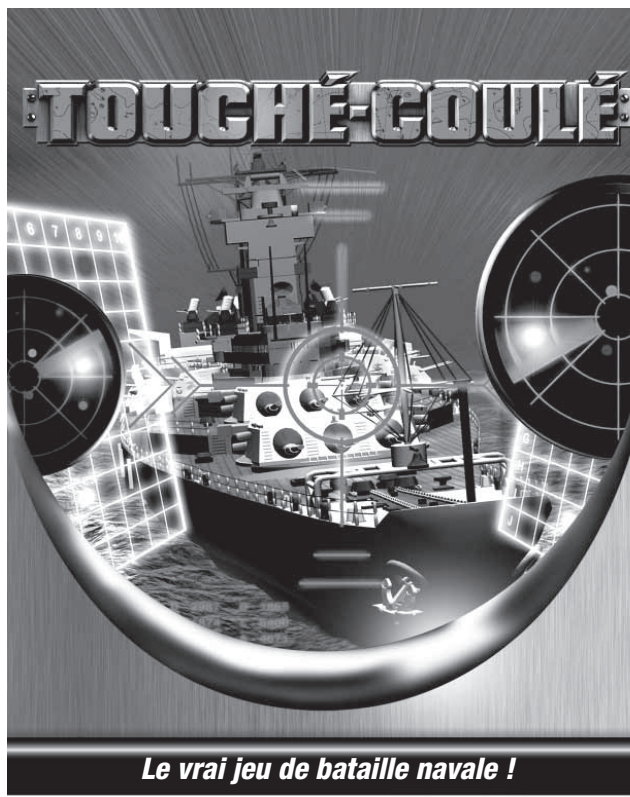
www.hasbro.fr

© 2005 Hasbro. Tous droits réservés.

Distribué en France par Hasbro France S.A.S. –  
Service Consommateurs : ZI Lourde, 57150 CREUTZWALD –  
Tél.: 08.25.33.48.85 – email : "conso@hasbro.fr".

Distribué en Belgique par S.A. Hasbro N.V.,  
't Hofveld 6 D, 1702 Groot-Bijgaarden.

Distribué en Suisse par / Vertriebs in der Schweiz durch / Distribuito in  
Svizzera da Hasbro Schweiz AG, Alte Bremgartenstrasse 2, CH-8965  
Berikon. Tel. 056 648 70 99.



**Le vrai jeu de bataille navale !**

**Contenu : 2 consoles de jeu, 2 flottes de 5 navires et  
des torpilles blanches et rouges.**

## LA MISSION

Être le premier joueur à couler tous les bateaux de son adversaire.

## PRÉPARATION DE LA BATAILLE

1. Sépare les consoles de jeu en les tirant de chaque côté.
2. Les deux adversaires s'installent face à face, leur console ouverte devant eux. Le couvercle de la console, appelé grille-cible, est ouvert et empêche ton adversaire de voir ta grille horizontale, appelée grille-océan.
3. Chaque joueur choisit 1 flotte de 5 navires et les détache de leur support (utiliser des ciseaux à bout rond en cas de difficulté).
4. Fixe tes navires dans les trous de la grille-océan, en les plaçant verticalement ou horizontalement (voir schéma 1). Attention : les navires ne doivent pas dépasser de la grille-océan.

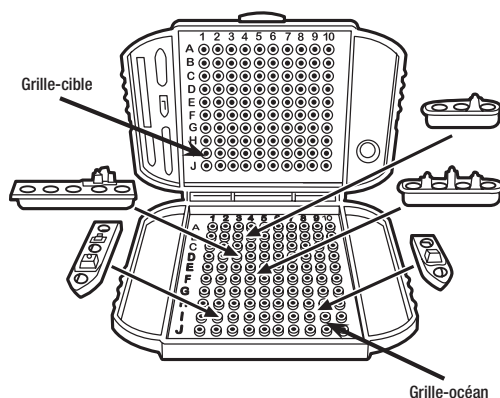


Schéma 1

## Chaque flotte est composée de :

Un porte-avions	(5 tirs)	
Un cuirassé	(4 tirs)	
Un croiseur	(3 tirs)	
Un sous-marin	(3 tirs)	
Un destroyer	(2 tirs)	

5. Chaque joueur se partage la moitié des torpilles rouges et des torpilles blanches et les place dans les compartiments sur les côtés de la grille-océan (voir schéma 2).

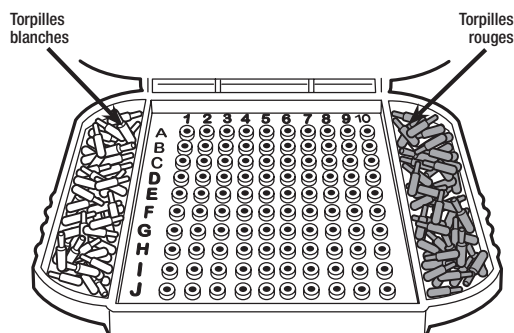


Schéma 2