
420-A11 | ADMINISTRATION

Projet Excel

TABLE DES MATIERES

Présentation générale	1
Contraintes communes	1
Projet 1 – Gestion d’horaire	2
Objectif général	2
Objectifs spécifiques	2
Notes techniques	3
Fichiers donnés	3
Captures d’écran	4
Projet 2 – Simulateur de gravité	6
Objectif général	6
Objectifs spécifiques	6
Notes importantes sur ce travail	7
Quelques notions et rappels mathématiques	7
Rappel sur la physique newtonienne	7
Rappels mathématiques	8
Note technique	9
Fichiers donnés	10
Captures d’écran	11
Projet 3 – Reconnaissance optique de caractères	13
Objectif général	13
Avant-propos	13
Objectifs spécifiques	13
Notes importantes sur ce travail	14
Quelques notions	14
Notes techniques	16
Fichiers donnés	16
Captures d’écran	17
Projet 4 – Bataille navale	19
Objectif général	19
Objectifs spécifiques	19
Notes importantes pour ce projet	19
Notes techniques	20
Capture d’écran	21

PRÉSENTATION GÉNÉRALE

Ce projet, qui est à remettre et qui sera évalué, consiste à réaliser un document *Excel* répondant à un mandat précis.

On vous propose quatre mandats différents et vous devez choisir le projet qui vous semble le plus intéressant. Les projets proposés sont tous différents et abordent des problématiques différentes. Ne vous fiez pas aux apparences, aucun des projets n'est facile et vous devez prévoir plusieurs heures de travail afin de réaliser un document de qualité.

Voici un résumé des trois projets :

- un outil de gestion d'horaire pour un étudiant;
- un simulateur de gravité pour un corps en mouvement;
- un outil de reconnaissance de caractères;
- un jeu de type bataille navale.

CONTRAINTES COMMUNES

Vous devez respecter ces contraintes pour chacun des projets :

- Les projets se font **individuellement**. L'entre-aide constructive est encouragée à condition que chaque étudiant maîtrise complètement ce qu'il présente. À tout moment et sans préavis, l'enseignant se réserve le droit de questionner un étudiant sur son travail. Un étudiant incapable d'expliquer une solution qu'il met de l'avant se verra attribuer une note de 0 pour plagiat.
- Contrairement aux exercices où la mise en forme était imposée, vous devez faire **votre propre mise en forme**. En fait, il est interdit de reprendre la même mise en forme que les exemples donnés par l'enseignant. Chaque étudiant doit présenter son travail de façon soignée et selon sa touche personnelle. Soyez informé que la qualité de la présentation compte de façon substantielle.
- Chacun des projets présente des problématiques et des difficultés différentes qui sont plus difficiles que les exercices habituels. C'est la responsabilité de l'étudiant d'**assister à la présentation que l'enseignant fera de chaque projet**. Ainsi, pour chaque projet l'enseignant présentera : une démarche de résolution, les concepts spécifiques sous-jacents à chaque projet, un rappel mathématique pour les projets le requérant et finalement la présentation de nouveaux outils d'*Excel* propre à chaque projet.
- Pour chacun des projets, on vous demande d'**ajouter un élément fonctionnel de votre cru** permettant d'améliorer l'application. L'ajout peut être de n'importe quel ordre, mais les points sont attribués sur la pertinence, le niveau de difficulté et la qualité de sa réalisation. Il est important de discuter avec l'enseignant afin de valider votre idée avant de la réaliser. À même le fichier *Excel* que vous allez remettre, vous devez présenter clairement quel est l'ajout personnel que vous avez fait. Sans cette courte présentation, l'ajout ne sera pas corrigé.
- Concernant les retards, une pénalité de 20% par jour de retard sera appliquée.
- Bien que les fichiers *Excel* donnés en exemple soient verrouillés, sachez qu'ils sont très faciles à déverrouiller. Pour cette raison, une vidéo vous est présentée afin d'illustrer le comportement du fichier *Excel* donné en exemple. Pendant les heures de cours, chaque étudiant peut essayer le fichier *Excel* de démonstration sur l'ordinateur de l'enseignant.

PROJET 1 – GESTION D’HORAIRE

OBJECTIF GÉNÉRAL

Vous devez réaliser un outil de gestion d’horaire pour un étudiant. Vous devez mettre en place un outil réparti sur trois feuilles :

1. un outil de saisie permettant de mettre en place tous les paramètres de l’horaire;
2. un outil de visualisation de la grille horaire;
3. les paramètres et données de référence de l’application.

OBJECTIFS SPÉCIFIQUES

Pour chaque feuille, voici ce qui est demandé :

- Feuille 1 – Planification de l’horaire
 - Vous devez mettre en place un outil permettant de définir tous les paramètres de chacun des cours dans l’horaire.
 - Vous devez offrir une possibilité de 9 cours par session.
 - Pour chacune des cours, voici les informations devant être utilisées (saisies manuellement par l’usager ou affichées automatiquement) :
 - Le sigle du cours *usager*
 - Le nom du cours *automatiquement*
 - Le nombre de périodes du cours *automatiquement*
 - Puisque les cours sont parfois donnés en deux périodes différentes dans la semaine, vous devez offrir deux groupes potentiels permettant de définir chacun des paramètres d’assignation de temps :
 - Jour *usager*
 - Heure de début du cours *usager*
 - Le nombre de périodes pour cette période *usager / automatiquement*
 - Heure de fin du cours *automatiquement*
 - Le local *usager*
 - Veuillez identifier clairement les cellules éditables et non éditables par un code de couleur (par exemple, toutes les cellules non éditables présentent un texte gris moyen).
- Feuille 2 – Présentation de l’horaire
 - La deuxième feuille a pour mandat de présenter l’horaire dans sa forme standard et synthétisée. Elle doit transformer toutes les informations linéaires de la planification de l’horaire en un affichage conventionnel sous forme de tableau 2D.
 - Il est essentiel que les conflits d’horaire apparaissent clairement.
 - Le contenu de cette feuille est complètement calculé automatiquement sans intervention de l’usager.
- Feuille 3 – Paramètres et références
 - Cette feuille doit contenir toutes les données nominales de l’application :
 - Les 3 paramètres de configuration de la grille horaire :
 - Heure de début de la journée (de la première période)
 - Durée d’une période de cours
 - Durée de l’intervalle entre chacune des périodes du cours

- La liste des locaux disponibles
- La liste des cours (incluant : sigle, nom, abréviation et nombre de périodes)
- Vous pouvez y ajouter les données propres à vos besoins.

NOTES TECHNIQUES

La fonction **EQUIV** permet de retrouver la position d'un élément désiré uniquement dans une plage linéaire (un tableau 1D). Il est fort probable que vous ayez besoin de trouver la position d'un élément se trouvant dans une plage 2D.

Étrangement, il n'existe pas de fonction dans Excel pour faire cette tâche. Néanmoins, on peut y arriver avec l'aide des fonctions matricielles que nous n'avons malheureusement pas eu le temps de voir en classe. Ces fonctions sont très puissantes et une courte présentation par l'enseignant vous permettra de comprendre leur pertinence et leur usage.

Afin d'utiliser ces types de fonctions, il est essentiel d'informer Excel qu'il doit basculer en mode matriciel pour réaliser les calculs demandés. Pour ce faire, lorsqu'on termine la saisie de la fonction dans la cellule, on appuie sur **CTRL+MAJ+ENTRÉE** ou lieu du simple **ENTRÉE** habituel.

L'usage des fonctions matricielles permet de retourner la position relative à une plage pour la première occurrence d'un élément recherché. Deux fonctions différentes doivent être créées pour extraire la ligne et la colonne.

Pour ces exemples de code, les mots en caractères gras indiquent une fonction alors que les mots en italiques indiquent une plage. Le mot *PLAGE* fait référence à la plage de données où doit s'effectuer la recherche. Le mot *VALEUR* fait référence à la valeur recherchée dans la plage.

Pour la ligne

```
= MAX( (PLAGE = VALEUR) * LIGNE(PLAGE) ) -  
MIN( LIGNE(PLAGE) ) + 1
```

Pour la colonne

```
= MAX( (PLAGE = VALEUR) * COLONNE(PLAGE) ) -  
MIN( COLONNE(PLAGE) ) + 1
```

Attention, pour que cette approche fonctionne bien, il faut que la matrice de données ne contienne qu'une seule instance de la valeur recherchée. Sinon, le résultat sera la première instance de la valeur recherchée indépendamment de la ligne et de la colonne.

FICHIERS DONNÉS

Deux exemples sont fournis. Le premier, plus simple, correspond à une version minimum à réaliser alors que le second présente une version plus sophistiquée avec plusieurs améliorations pouvant servir d'inspiration.

Un fichier Excel contenant une liste de cours vous évitera de tout retaper.

CAPTURES D'ÉCRAN

Planification de l'horaire

Planification de l'horaire

Sélection des cours				Planification de la 1 ^{ère} période					Planification de la 2 ^{ème} période				
No	Cours	Nom	Périod	Jour	Heure	Nb.	Heure fin	Local	Jour	Heure	Nb.	Heure fin	Local
1	601-101-MQ	Écriture et littérature	4	Lundi	8:00	2	9:45	4.08	Mercredi	8:55	2	10:40	3.34
2	604-101-MQ	Langue anglaise et communication	3	Jeudi	11:40	3	14:20	5.29					
3	109-114-MQ	Randonnée pédestre (255)	2	Lundi	10:45	2	12:30	6.66					
4	201-A12-VM	Mathématique pour l'informatique 1	3	Jeudi	8:00	3	10:40	4.08					
5	410-A11-VM	Administration	5	Vendredi	8:00	3	10:40	5.29	Mardi	15:20	2	17:05	3.34
6	420-B11-VM	Programmation structurée	6	Mardi	8:00	3	10:40	6.66	Mercredi	11:40	3	14:20	4.08
7	420-B12-VM	Outils et matériels informatiques	6	Lundi	15:20	3	18:00	5.29	Jeudi	14:25	3	17:05	5.29
8	340-101-MQ	Philosophie et rationalité	4	Mercredi	14:25	2	16:10	6.66	Vendredi	15:20	2	17:05	3.34
9	601-102-MQ	Littérature et imaginaire	4	Vendredi	13:30	2	15:15	6.45	Mardi	11:40	2	13:25	3.34

Légende

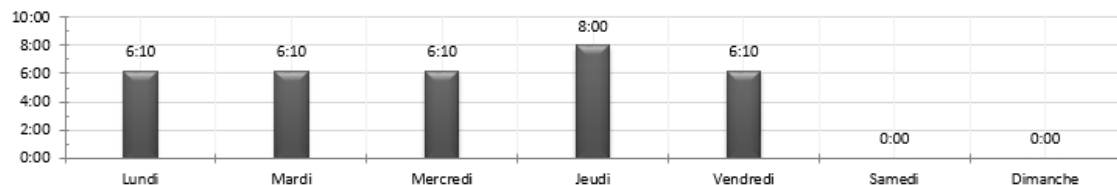
Aucun cours
abc Planification adéquate
abc Planification incomplète
abc Planification dépassant la plage horaire de fin de journée
abc Conflit d'horaire

Feuille 1 – Planification d'horaire

Présentation de l'horaire

Présentation de l'horaire

		Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi	Dimanche
1	8:00	601-101-MQ	420-B11-VM	-	201-A12-VM	410-A11-VM	-	-
2	8:55	Local 4.08	Prog. structurée	601-101-MQ	Math. pour info. 1	Administration	-	-
3	9:50	-	Local 6.66	Local 3.34	Local 4.08	Local 5.29	-	-
4	10:45	109-114-MQ	-	-	-	-	-	-
5	11:40	Local 6.66	601-102-MQ	420-B11-VM	604-101-MQ	-	-	-
6	12:35	-	Local 3.34	Prog. structurée	Langue angl. et com.	-	-	-
7	13:30	-	-	Local 4.08	Local 5.29	601-102-MQ	-	-
8	14:25	-	-	340-101-MQ	420-B12-VM	Local 6.45	-	-
9	15:20	420-B12-VM	410-A11-VM	Local 6.66	Outils et mat. info.	340-101-MQ	-	-
10	16:15	Outils et mat. info.	Local 3.34	-	Local 5.29	Local 3.34	-	-
11	17:10	Local 5.29	-	-	-	-	-	-



Nombre d'heures par jour

Feuille 2 – Présentation de l'horaire

Paramètres et références

Paramètres et références

Paramètres de la plage horaire		Plage horaire		Jour de la semaine		Liste des locaux		Liste des cours et informations relatives				
Heure de la première période	08:00	1	8:00 8:50	1	Lundi		4.07	Session 1	Code	Nom	Abbréviation	Nombre de périodes
Durée de chaque période	00:50	2	8:55 9:45	2	Mardi		4.08		601-101-MQ	Écriture et littérature	Écriture et lit.	4
Interval entre chaque période	00:05	3	9:50 10:40	3	Mercredi		5.29		604-101-MQ	Langue anglaise et communication	Langue angl. et com.	3
		4	10:45 11:35	4	Jeudi		3.34		109-114-MQ	Randonnée pédestre (255)	Randonnée pédestre	2
		5	11:40 12:30	5	Vendredi		6.66		201-A12-VM	Mathématique pour l'informatique 1	Math. pour info. 1	3
		6	12:35 13:25	6	Samedi		7.15		410-A11-VM	Administration	Administration	5
		7	13:30 14:20	7	Dimanche		6.45	420-B11-VM	Programmation structurée	Prog. structurée	6	
		8	14:25 15:15					420-B12-VM	Outils et matériels informatiques	Outils et mat. info.	6	
		9	15:20 16:10					340-101-MQ	Philosophie et rationalité	Philo. et rationalité	4	
		10	16:15 17:05					601-102-MQ	Littérature et imaginaire	Littérature et imagin	4	
		11	17:10 18:00					604-4T1-VM	Anglais des techniques I	Anglais des tech. 1	3	
								109-213-MQ	Démarche et efficacité en power yoga	Démarche power Yog	2	
								201-A22-VM	Mathématique pour l'informatique II	Math. pour info. 2	4	
								420-B21-VM	Programmation avancée	Prog. avancée	6	
								420-B22-VM	Systèmes d'exploitation	Système d'exploitatio	6	
								340-102-MQ	L'être humain	L'être humain	3	
								601-103-MQ	Littérature québécoise	Littérature québécois	4	
								109-311-MQ	Mouvement / relaxation et autonomie	Mouv. relax. auto.	2	
								350-A32-VM	Communications et interrelations	Comm. et interrela.	3	
								420-B31-VM	Structures de données	Structures de donnée	6	
								420-B32-VM	Administration Linux et Windows	Admin. Linux et Win.	5	
								420-B33-VM	Concepts de l'orienté objet	Orienté objet	6	
								340-4EB-VM	Éthique et politique	Éthique et politique	3	
								601-4EB-VM	Littérature et culture contemporaines	Litt. et cult. contemp	4	
								202-5DS-VM	500 ans de développements scientifiques	Dév. scientifique	3	
								201-A31-VM	Traitement statistique des données	Traitement stat.	3	
								420-B41-VM	Gestion de projet: Techniques et méthodologie	Gestion de projet	5	
								420-B42-VM	Langages d'exploitation des bases de données	Langage d'expl. des BI	5	
								420-B43-VM	Programmation par objets avancée	Prog. objet avancée	6	
								101-NHU-VM	Nutrition humaine	Nutrition humaine	3	

Feuille 3 – Paramètres et références

PROJET 2 – SIMULATEUR DE GRAVITÉ

OBJECTIF GÉNÉRAL

Ce projet consiste à faire la simulation des effets d'attraction/répulsion qu'exercent trois corps statiques sur un corps dynamique.

Vous devez mettre en place un outil réparti sur deux feuilles présentant ceci :

- une zone de saisie permettant de déterminer les conditions initiales ainsi qu'un graphique représentant la trajectoire obtenue;
- une simulation animée de la trajectoire dans le temps.

OBJECTIFS SPÉCIFIQUES

Pour chaque feuille, voici ce qui est demandé :

- Feuille 1 – Conditions initiales et trajectoire
 - Vous devez mettre en place un outil permettant de définir tous les paramètres des conditions initiales. Certains paramètres doivent être définis par l'utilisateur alors que d'autres sont calculés automatiquement :
 - Paramètres généraux :

• base de temps	t	<i>usager</i>
• ratio de la constante d'Aarseth	A	<i>usager</i>
 - Les paramètres du corps en mouvement :

• la masse	m	<i>usager</i>
• la position initiale en x et y	p	<i>usager</i>
• la vitesse initiale en x et y	v	<i>usager</i>
• l'accélération initiale en x et y	a	<i>usager</i>
 - Les paramètres pour chacun des trois corps statiques

• la position en x et y	q	<i>usager</i>
• le rayon de la sphère représentant le corps	r	<i>usager</i>
• la masse volumique moyenne du corps	ρ	<i>usager</i>
• le volume	V	<i>automatiquement</i>
• la masse totale	M	<i>automatiquement</i>
• la constante d'Aarseth	λ	<i>automatiquement</i>
• le choix entre un corps attractif et répulsif	ω	<i>usager</i>
 - Vous devez aussi inclure un graphique affichant ceci :
 - un système d'axe affichant les dimensions;
 - trois cercles indiquant les trois corps statiques;
 - la trajectoire obtenue.
- Feuille 2 – Simulation en mouvement
 - le graphique animé;
 - un bouton permettant d'amorcer l'animation;
 - un petit texte indiquant l'état d'avancement de la trajectoire en pourcentage.

NOTES IMPORTANTES SUR CE TRAVAIL

Cette simulation représente le déplacement d'un corps dynamique subissant les effets des forces appliquées par trois corps statiques. L'approche présentée ici repose sur des approximations faibles et met l'accent davantage sur le résultat graphique que sur une représentation physique exacte. D'un point de vue théorique, cette simulation est très simpliste, mais permet un rendu visuel tout de même intéressant.

Plusieurs simplifications sont faites dans le but d'alléger la simulation. Principalement, on note les points suivants :

- les déplacements sont calculés par des pas d'intégration discrets (ignorant la nature réelle du mouvement entre ces pas);
- les équations de mouvement sont basées sur la théorie newtonienne (l'étudiant chevronné pourra se référer aux équations d'Einstein);
- il n'y a aucune détection et gestion des collisions (implique que les corps restent intègres en passant les uns au travers les autres);
- la constante d'Aarseth est intégrée au modèle afin de minimiser les accélérations trop brusques près des points de singularité;
- le coefficient de friction (présenté dans le second exemple) est une constante fixe ne tenant pas compte de la vitesse ou d'autres facteurs;
- on vous demande de générer 1000 points de trajectoire dans le temps.

QUELQUES NOTIONS ET RAPPELS MATHÉMATIQUES

Le problème à N corps consiste à déterminer la position des corps interagissant gravitationnellement les uns avec les autres, connaissant leurs masses, leurs positions, vitesses et accélérations initiales. Il s'agit d'un problème mathématique fondamental pour la physique et un défi significatif pour les domaines de la mathématique symbolique et numérique. Le problème à N corps se pose dans le cadre newtonien et peut devenir rapidement très difficile. Néanmoins, cette étude se pose aussi dans le cadre de la relativité générale et devient largement plus complexe.

RAPPEL SUR LA PHYSIQUE NEWTONNIENNE

Le calcul de la position courante en fonction des valeurs de position, vitesse et accélération précédente :

$$\vec{p}_i = \vec{p}_{i-1} + \vec{v}_{i-1}\Delta t + \vec{a}_{i-1}\Delta t^2$$

les unités de mesure
 $m = m + (m/t * t) + (m/t^2 * t^2)$
 $m = m + m + m$

Le calcul de la vitesse courante en fonction de la vitesse et de l'accélération précédente :

$$\vec{v}_i = \vec{v}_{i-1} + \vec{a}_{i-1}\Delta t$$

La sommation se fait à chaque planète
Puisque le vecteur représente en X et Y, c'est 2
On a 6 colonnes de calculs

Le calcul de l'accélération courante de n corps gravitant les uns autour des autres :

$$\vec{a}_{ji} = \sum_{\substack{k=1 \\ k \neq j}}^n m_k \frac{\vec{p}_{ki} - \vec{p}_{ji}}{(|\vec{p}_{ki} - \vec{p}_{ji}|^2 + \lambda_k^2)^{\frac{3}{2}}}$$

$|\vec{p}_{ki} - \vec{p}_{ji}|^2$ = la partie pythagore

Le travail demandé consiste à appliquer la force de trois corps statiques sur un seul corps en mouvement. Ainsi et considérant la dernière équation, l'accélération de l'unique corps en mouvement se simplifie et se complète par :

$$\vec{a}_i = \sum_{k=1}^3 m_k \frac{\vec{q}_k - \vec{p}_i}{(|\vec{q}_k - \vec{p}_i|^2 + \lambda_k^2)^{\frac{3}{2}}}$$

$$\vec{a}_i = m_1 \frac{\vec{q}_1 - \vec{p}_i}{(|\vec{q}_1 - \vec{p}_i|^2 + \lambda_1^2)^{\frac{3}{2}}} + m_2 \frac{\vec{q}_2 - \vec{p}_i}{(|\vec{q}_2 - \vec{p}_i|^2 + \lambda_2^2)^{\frac{3}{2}}} + m_3 \frac{\vec{q}_3 - \vec{p}_i}{(|\vec{q}_3 - \vec{p}_i|^2 + \lambda_3^2)^{\frac{3}{2}}}$$

Précisions sur les symboles utilisés :

Symbole	Signification
$\dots i$	au temps i
$\dots i - 1$	au temps $i - 1$, c'est-à-dire au temps précédent le temps i
Δt	l'intervalle de temps écoulé entre le temps i et le temps $i + 1$
p_i	la position du corps en mouvement au temps i
v_i	la vitesse du corps en mouvement au temps i
a_i	l'accélération du corps en mouvement au temps i
q_k	la position du k^e corps statique
m_k	la masse du k^e corps statique
λ_k	la constante d'Aarseth du k^e corps statique

RAPPELS MATHÉMATIQUES

Voici quelques rappels au niveau des calculs nécessaires pour ce projet.

Pour ce projet, nous ne considérons qu'un monde en 2D (la représentation 3D serait difficile avec Excel). Ainsi, la notation mathématique utilisée utilise une syntaxe vectorielle 2D. N'hésitez pas à poser des questions à votre enseignant si certaines formules vous causent des problèmes.

Symbole	Signification
\vec{u}	Cette syntaxe indique que u est un vecteur représentant deux valeurs à la fois, les composantes x et y de l'espace 2D. $\vec{u} = \begin{bmatrix} u_x \\ u_y \end{bmatrix}$
$ \vec{u} $	Cette notation s'appelle la norme du vecteur (ou la longueur du vecteur). On utilise ici simplement la distance euclidienne définie par : $ \vec{u} = \sqrt{u_x^2 + u_y^2}$
V	Chaque corps étant représenté par une sphère, on calcule son volume par (où r est le rayon du corps) : $V = \frac{4}{3}\pi r^3$
M	La masse des corps se calcul simplement par ceci (où ρ est la masse volumique du corps) : $M = \rho V$
λ	La constante d'Aarseth est donnée par une valeur proportionnelle au rayon du corps concerné (où Λ est le ratio de la constante d'Aarset) : $\lambda = r\Lambda$

NOTE TECHNIQUE

Excel n'offre aucun outil permettant d'animer directement un graphique. Il est donc nécessaire de programmer un tel outil en tirant avantage de la façon dont il gère les graphiques. Même si nous n'avons pas abordé le sujet en classe, il faut utiliser un peu de *VBA* (*Visual Basic for Application*) pour y arriver. L'idée est relativement simple et se fait en 5 étapes :

1. créer d'abord toutes les données devant être affichées dans le graphique (on précise qu'il faut créer toutes les données dès le départ pour chaque étape d'animation – ainsi, à chaque étape, on récupère les données précalculées et on les présente au graphique);
2. ensuite, déterminer trois cellules contenant les indexes de l'animation (qui pourront être masquées au besoin) :
 - a. position initiale;
 - b. position finale;
 - c. position courante;
3. utiliser les fonctions de recherche (telles que **INDEX** et **EQUIV**) afin d'extraire les données calculées en 1 de la i^{e} étape d'animation (correspondant à la position courante i);
4. créer un graphique utilisant la plage de données faite en 3;
5. finalement, écrire un petit script en *VBA* réalisant ces étapes :

- a. une boucle changeant la valeur de la position courante en partant de la position initiale jusqu'à la position finale;
- b. recalculer l'étape 3;
- c. effectuer le rafraichissement du graphique.

Voici le code VBA nécessaire :

```
' -----
' VBA - Les commentaires débutent par une apostrophe
' -----
' Changer les champs suivants pour vos cellules et plages correspondantes :
' - A1 = La cellule où se trouve la position initiale
' - A2 = La cellule où se trouve la position finale
' - A3 = La cellule réceptrice de la valeur courante de la position
' - J1:M10 = La plage des cellules à recalculer pour le graphique
' - GraphToAnimate = Le nom du graphique (voir l'onglet Disposition du graphique)
' -----

Option Explicit
Option Base 0

' -----
' Ici est la fonction qui sera créée par le bouton
Private Sub CommandButton1_Click()
    Startsimulation
End Sub

' -----
Sub Startsimulation()
    Dim i As Integer
    For i = Range("A1").Value To Range("A2").Value
        Range("A3").Value = i
        Application.ActiveSheet.Range("J1", "M10").Calculate
        ActiveSheet.ChartObjects("GraphToAnimate").Application.Calculate
        'WaitFor (1)
    Next i
End Sub

' La fonction suivante sert à ralentir Excel lors de l'animation.
' Néanmoins, Excel est tellement lent que cette fonction n'est pas nécessaire.
Sub WaitFor(sec)
    Dim rightNow As Date
    rightNow = Now()
    Application.Wait TimeSerial(Hour(rightNow), Minute(rightNow), second(rightNow) + sec)
End Sub
```

Finalement, voici les étapes pour insérer un bouton sur l'interface et créer le lien entre le bouton et le code VBA qu'il doit exécuter :

1. assurez-vous que l'onglet *Développeur* est affiché sur le ruban (voir : *Fichier – Options – Personnaliser le ruban* – à droite, activer l'onglet);
2. sur l'onglet *Développeur*, dans le groupe *Contrôle*, Insérer un *Contrôle ActiveX* de type *Bouton de commande* et dessinez-le directement sur la feuille *Excel* (vous remarquerez que le *Mode Création* vient d'être activé – voir le groupe *Contrôle* de l'onglet *Développeur*);
3. double cliquez sur le bouton et automatiquement apparaît l'interface de travail de *Microsoft Visual Basic pour Application (VBA)*, on remarque qu'on a déjà une fonction de définie qui s'intitule : `Private Sub CommandButton1_Click()`
4. quitter le *Mode Création*.

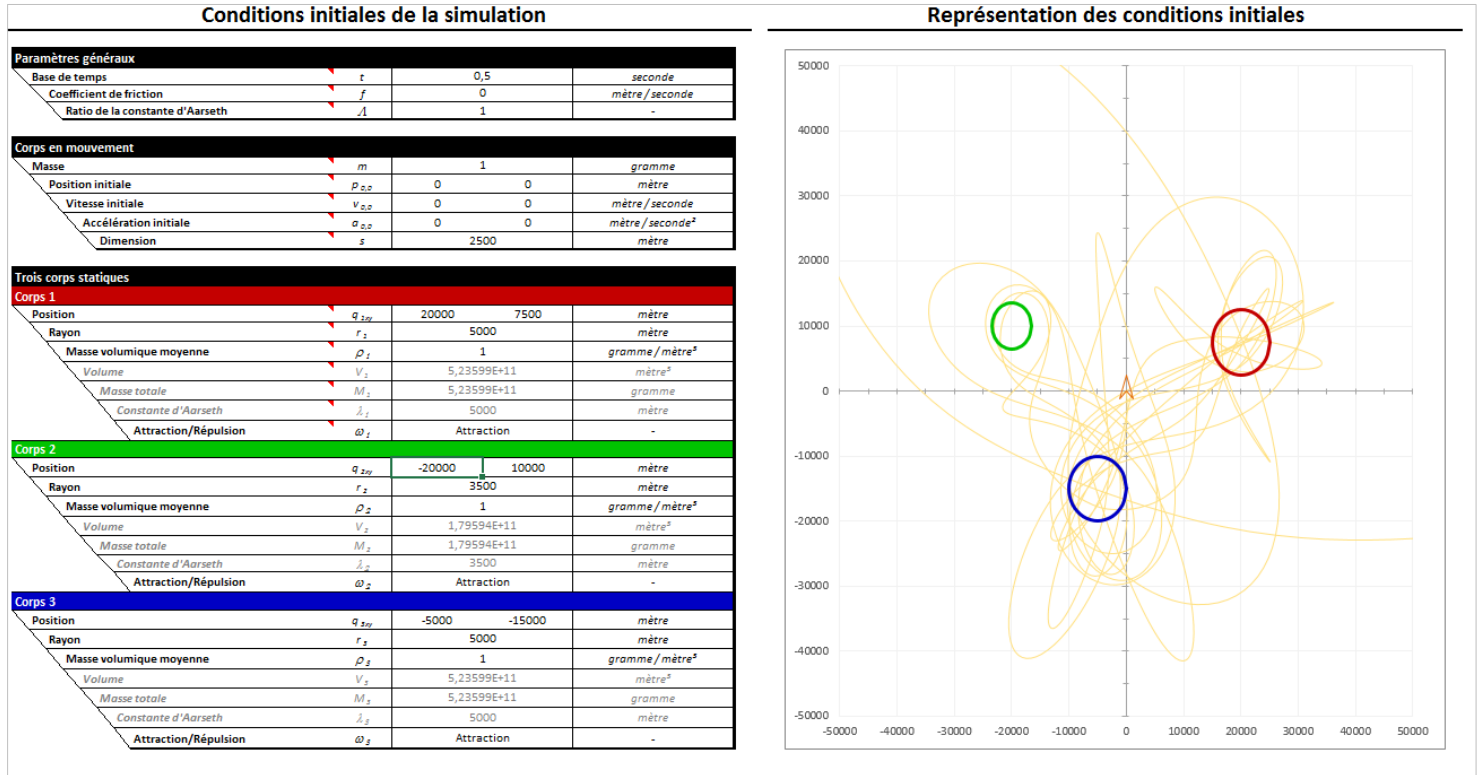
Finalement, lorsque votre fichier *Excel* comporte une macro ou du code, ce dernier doit être sauvegardé avec un nouveau format de fichier nommé *XLSM* plutôt que *XLXS*.

FICHIERS DONNÉS

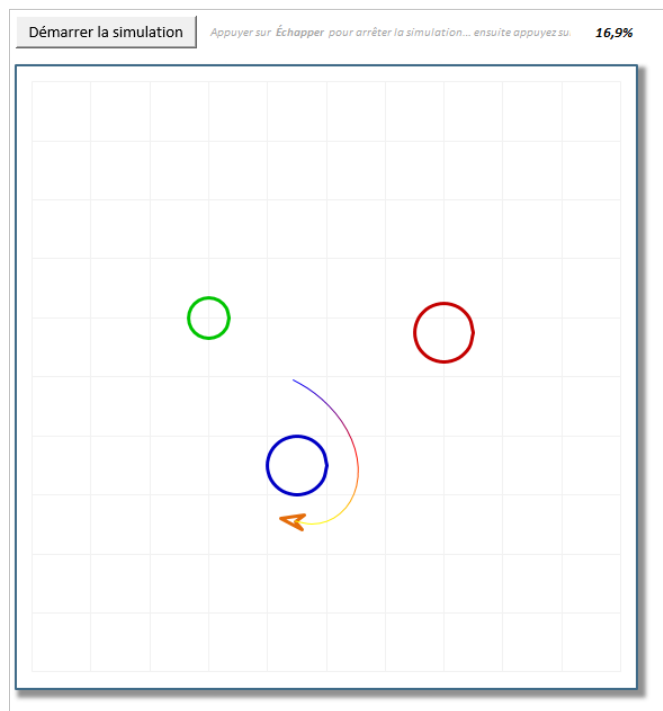
Trois exemples sont fournis. Le premier, plus simple, correspond à une version minimum à réaliser alors que le second présente une version plus sophistiquée pouvant servir d'inspiration. Finalement, le troisième exemple présente une solution que seuls les étudiants les plus chevronnés pourront tenter où 12 corps gravitent les uns autour des autres.

Un fichier Excel est donné et présente en détail comment faire l'animation d'un graphique.

CAPTURES D'ÉCRAN



Feuille 1 – Conditions initiales et trajectoire



Feuille 2 – Simulation en mouvement

PROJET 3 – RECONNAISSANCE OPTIQUE DE CARACTÈRES

OBJECTIF GÉNÉRAL

Ce projet consiste à faire un outil simpliste de reconnaissance optique de caractères (ROC) communément appelé OCR (en anglais « *Optical Character Recognition* »).

Vous devez mettre en place un outil réparti sur trois feuilles présentant ceci :

- une zone de saisie permettant de déterminer les caractères à reconnaître;
- une zone de saisie permettant de déterminer les caractères de référence qui serviront de base de reconnaissance;
- les résultats de la reconnaissance.

AVANT-PROPOS

La réalisation de ce projet dans *Excel* requiert la mise en place de plage simulant les images de caractères. Pour y arriver, on doit déterminer une taille commune pour chaque caractère. L'exemple proposé utilise une taille de 7 x 9 pixels, mais vous pouvez utiliser votre propre taille. Sachez seulement que l'approche utilisée est très sensible à la résolution et qu'une taille de 7 x 9 est déjà très petite. Idéalement, vous ne devriez pas réduire cette taille. Le code VBA fourni n'a pas été validé pour des tailles différentes, mais devrait fonctionner. Si des modifications sont nécessaires, elles ne devraient pas être très significatives et l'enseignant pourra vous aider à ce niveau.

Pour chacun des caractères, on vous demande de simuler des images binaires où seules les valeurs 0 et 1 sont utilisées pour chacun des pixels (0 pour le fond et 1 pour le signal). L'étudiant plus chevronné pourra aborder la problématique de segmentation en mettant des pixels en niveaux de gris ou même en couleur.

La création de contenus (les caractères) pour ce projet est une tâche fastidieuse dans *Excel*. C'est pourquoi du code VBA vous est donné afin de faciliter trois tâches : effacer le contenu des plages, enregistrer le contenu des plages dans un fichier texte et ouvrir un fichier texte. Sachez que les fonctions d'ouverture de fichier ne font pas de validation des fichiers spécifiés. Ainsi, si vous ouvrez un fichier ne correspondant pas au format prescrit, *Excel* arrêtera l'exécution du script VBA. Le format est simple et peut facilement être compris et modifié en regardant un fichier créé par les fonctions de sauvegarde.

OBJECTIFS SPÉCIFIQUES

Pour chaque feuille, voici ce qui est demandé :

- Feuille 1 – Phrase à reconnaître
 - Vous devez mettre en place la définition d'une phrase devant être reconnue. Cette phrase doit posséder au moins 10 caractères.
 - Les caractères doivent se mettre en couleur automatiquement en fonction des valeurs inscrites (0 ou 1 pour une image binaire).
 - Vous devez faire l'intégration du code VBA permettant d'effacer, d'enregistrer et d'ouvrir une série de caractères.
- Feuille 2 – Caractères de référence

- Vous devez mettre en place la définition d'au moins 12 caractères de référence. Ces caractères serviront de base de comparaison lors de la reconnaissance finale.
- Vous devez en plus avoir une zone indiquant quel est le caractère représenté par l'image.
- Les caractères doivent se mettre en couleur automatiquement en fonction des valeurs inscrites (0 ou 1 pour une image binaire).
- Vous devez faire l'intégration du code VBA permettant d'effacer, d'enregistrer et d'ouvrir une série de caractères.
- L'exemple donné présente l'option d'afficher ou non les centroïdes pour chaque caractère. Cet affichage est optionnel.
- Feuille 3 – Reconnaissance
 - Vous devez mettre en place une façon de présenter les résultats de la reconnaissance.
 - Vous devez afficher :
 - la phrase à reconnaître (sous forme d'image);
 - les caractères reconnus et non reconnus (sous forme d'image);
 - la phrase finale reconnue (sous forme de texte).
 - Vous devez laisser à l'utilisateur de définir le seuil de distance minimale acceptée pour la reconnaissance. L'exemple donné utilise une barre de défilement pour cette tâche. L'usage d'un tel contrôle est optionnel et il est possible d'utiliser uniquement une cellule standard.

NOTES IMPORTANTES SUR CE TRAVAIL

La reconnaissance de caractères est une tâche complexe qui dépasse largement le cadre du cours (même d'un cours universitaire de premier cycle). Néanmoins, l'approche simpliste présentée ici utilise des outils de base communément utilisés pour ce défi. Quoiqu'incomplète, cette introduction permet de comprendre une partie du traitement requis pour cette tâche.

En temps normal, les étapes sont :

1. la numérisation;
2. la filtration;
3. la segmentation;
4. descripteurs de formes;
5. la reconnaissance.

Pour ce projet, seules les deux dernières tâches sont traitées.

Finalement, on simplifie le calcul des descripteurs de formes à quelques outils mathématiques se calculant simplement avec Excel. Le taux de reconnaissance obtenu n'est pas très robuste, mais permet tout de même d'obtenir un résultat satisfaisant.

QUELQUES NOTIONS

Pour ce projet, on utilise une seule famille de descripteurs de formes : les moments d'inertie d'ordre inférieur. Ces descripteurs sont souvent de bons indicateurs et servent aussi à construire d'autres descripteurs plus performants.

D'abord, pour chaque image, on calcule les moments d'inertie standard notés M_{pq} :

$$M_{pq} = \sum_x \sum_y x^p y^q I_{xy}$$

Cette notation implique le calcul de plusieurs moments d'inertie standard. Le tableau présenté plus loin montre tous les moments M_{pq} à calculer.

À l'aide de M_{00} , M_{10} et M_{01} on calcule les centroïdes en x et en y notés respectivement \bar{x} et \bar{y} .

$$\bar{x} = \frac{M_{10}}{M_{00}}$$

$$\bar{y} = \frac{M_{01}}{M_{00}}$$

Il est possible de rendre invariants à la translation les moments standards en calculant les moments d'inertie centrés notés μ_{pq} :

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I_{xy}$$

Finalement, il est possible de rendre ces moments invariants à la taille par les moments d'inertie centrés réduits η_{pq} :

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\left(1 + \frac{p+q}{2}\right)}}$$

Symbole	Signification
M_{pq}	moments d'inertie standards $2d$ d'ordre p et q
μ_{pq}	moments d'inertie centrés $2d$ d'ordre p et q
η_{pq}	moments d'inertie centrés réduits $2d$ d'ordre p et q
\bar{x}	le centroïde en x
\bar{y}	le centroïde en y
I_{xy}	valeur du pixel à la position (x, y)
p	indice du moment de la dimension x
q	indice du moment de la dimension y

Le tableau suivant présente les différents moments d'inertie à calculer.

Moments	Ordres des moments à calculer
M_{pq}	M_{00} , M_{10} et M_{01}
μ_{pq}	μ_{00} , μ_{20} , μ_{02} , μ_{11} , μ_{30} , μ_{03} , μ_{21} et μ_{12} (sachez que $\mu_{00} = M_{00}$)
η_{pq}	η_{20} , η_{02} , η_{11} , η_{30} , η_{03} , η_{21} et η_{12}

Finalement, la reconnaissance se fera en calculant la distance euclidienne à 7 dimensions sur les moments centrés réduits par :

$$d = \sqrt{\sum_{k \in pq \mid pq = \{20,02,11,30,03,21,12\}} (\eta_k - \dot{\eta}_k)^2}$$

$$d = \sqrt{(\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2 + (\eta_{20} - \dot{\eta}_{20})^2}$$

où η_{pq} représente le caractère à reconnaître et $\dot{\eta}_{pq}$ représente le caractère de référence en comparaison.

Pour reconnaître un caractère, on doit calculer la distance entre ce caractère et chacun des caractères de référence (pour notre exemple, 12 distances doivent être calculées pour chacun des caractères à reconnaître). Il suffira de sélectionner le caractère ayant la distance la plus courte. Évidemment, un seuil doit être appliqué afin de déterminer la distance minimum à considérer.

NOTES TECHNIQUES

Même s'ils ne sont pas essentiels, les outils de calcul matriciel disponibles avec *Excel* sont d'une grande aide pour ce projet. Assurez-vous d'assister à la présentation de l'enseignant à ce sujet.

Par ailleurs, ce projet est grandement simplifié par l'usage de fonction de manipulation de référence. Assurez-vous de bien comprendre les fonctions : **index**, **equiv**, **adresse**, **décalage** et **indirect**. Encore une fois, l'enseignant fera une présentation de ces outils.

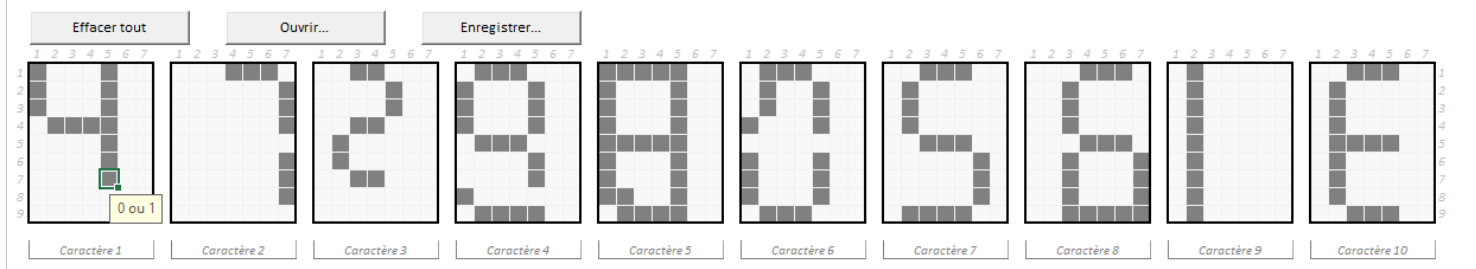
Une présentation de l'enseignant permettra de bien comprendre comment intégrer le code *VBA*.

FICHIERS DONNÉS

Plusieurs fichiers sont donnés pour ce projet :

- une vidéo présentant l'exemple;
- un fichier *Excel* présentant le code *VBA* à inclure;
- 4 fichiers texte contenant 12 caractères de référence de taille 7 x 9 pixels
CaracteresDeReference_*.txt;
- 4 fichiers texte contenant une phrase à reconnaître de 10 caractères de taille 7 x 9 pixels
PhraseAReconnaitre_*.txt.

Phrase à reconnaître



Feuille 1 – Phrase à reconnaître

Caractères de référence



Feuille 2 – Caractères de référence

Reconnaissance

Phrase à reconnaître

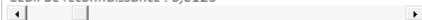
4 7 2 9 8 0 5 6 1 E



Résultat de la reconnaissance

4 7 2 9 8 0 5 6 1 E

Seuil de reconnaissance : 0,0125



472980561E

Feuille 3 – Reconnaissance

PROJET 4 – BATAILLE NAVALE

OBJECTIF GÉNÉRAL

Ce projet consiste à réaliser un jeu de type « *Battleship* » pour deux joueurs.

Vous devez mettre en place une grille de jeu répartie sur trois feuilles :

- une feuille expliquant les règles générales ainsi que les instructions d'utilisation de votre jeu;
- une feuille permettant de déterminer les paramètres généraux;
- une grille de jeu.

OBJECTIFS SPÉCIFIQUES

Pour chaque feuille, voici ce qui est demandé :

- Feuille 1 – Présentation
 - Vous devez présenter votre jeu en trois parties :
 - une présentation générale du jeu;
 - les règles du jeu;
 - les particularités de votre scénario de jeu ainsi que de votre implémentation.
- Feuille 2 – Paramètres
 - Vous devez être capable de saisir les informations générales nécessaires au jeu, notamment :
 - le nom de votre flotte de combat;
 - le nom du fichier de votre ennemi (incluant le chemin d'accès si requis);
 - identifier le joueur qui fera le premier tir.
- Feuille 3 – La grille de jeu
 - La grille de jeu consiste en 5 zones différentes :
 - le nom des deux participants au combat selon la forme : *Joueur 1 VS Joueur 2*;
 - une zone d'information affichant l'état d'avancement du jeu;
 - une petite zone escamotable affichant la légende;
 - une grille permettant de définir où sont les navires;
 - deux grilles permettant de faire le suivi de l'évolution de la partie (une grille pour le suivi des attaques ennemies et une grille pour le suivi de ses attaques).

NOTES IMPORTANTES POUR CE PROJET

Ce projet est particulier car il demande de créer un scénario de jeu interactif à deux joueurs « réseau » sans l'utilisation du VBA. Pour y arriver, on utilise la possibilité qu'Excel offre de faire des références aux cellules d'une feuille provenant d'un autre fichier. De cette façon, on doit créer **un** fichier qui sera copié en deux exemplaires. Chaque joueur utilisant sa propre copie du fichier indique, parmi les paramètres, quel est le fichier utilisé par l'adversaire. De cette façon, il est possible de créer un jeu interactif pourvu que chaque joueur ait accès au fichier ennemi.

Sans l'utilisation de VBA, il est impossible d'interdire des entrées de données non autorisées par le scénario d'un jeu tel qu'un logiciel le permet habituellement. Par exemple, il est impossible d'interdire la saisie d'un tir du joueur 1 alors que c'est au joueur 2 de jouer. Néanmoins, il est possible de faire un suivi complet de l'évolution de la partie et d'afficher l'état courant du jeu selon les conditions présentes. Ainsi, il devient possible d'informer les joueurs si une inconsistance survient.

Pour ce jeu, vous devez être en mesure d'identifier tous les cas de figure suivants et d'informer adéquatement les deux usagers :

- aucune assignation du nom fichier identifiant l'adversaire;
- le fichier ennemi identifié ne correspond pas au fichier du jeu (ce n'est pas un fichier *Excel* ou le fichier *Excel* n'est pas un fichier du jeu *Battleship*);
- le fichier ennemi identifié est « lui-même », autrement dit, le fichier pointe vers lui-même et tous les tirs seront vers soi-même;
- chaque joueur doit assigner le joueur qui débute la partie (les deux choix offerts sont *Moi* ou *L'autre* – on doit valider que chaque joueur est cohérent puisque l'un doit indiquer *Moi* alors que l'autre indique *L'autre*);
- chaque joueur doit s'assigner un nom (on doit valider que les noms sont différents pour chaque joueur);
- on doit identifier le moment où la flotte n'est pas complètement construite (alors que les navires ne sont pas complets et tous présents sur la grille de la zone de construction);
- on doit aussi identifier lorsqu'un navire n'est pas construit selon les règles, c'est-à-dire lorsqu'il n'est pas disposé sur des cases contiguës alignées verticalement ou horizontalement exclusivement;
- lorsque la partie est en cours, on doit identifier le joueur qui doit jouer et informer les joueurs en conséquence (le joueur devant jouer est invité à définir son coup alors que l'autre est invité à attendre le coup adverse);
- on doit être en mesure d'identifier s'il y a un vainqueur et, si la partie est terminée, on doit afficher le résultat aux deux joueurs;
- on doit identifier tous les états inconsistants de jeu et informer les joueurs (principalement si un joueur a joué alors que c'était à son adversaire);
- et finalement tout autre état pertinent pour votre implémentation.

NOTES TECHNIQUES

La syntaxe permettant de faire référence à un élément d'un fichier externe est :

```
= '[chemin d acces\nom de fichier]nom de feuille'!référence
```

par exemple :

```
= '[c:\a11\projet\battleship.xlsx]Bataille navale'!&A1:B&10
```

Ce projet est grandement simplifié par l'usage de fonction de manipulation de référence. Assurez-vous de bien comprendre les fonctions : **index**, **equiv**, **adresse**, **décalage** et **indirect**. Encore une fois, l'enseignant fera une présentation de ces outils.

CAPTURE D'ÉCRAN

Battleship

Règle du jeu

- Le jeu de bataille navale (en anglais *BattleShip*) est un jeu de société dans lequel deux joueurs doivent placer des navires sur une grille tenue secrète et tenter de couler les navires adverses.
- Le gagnant est celui qui parvient à torpiller complètement les navires de l'adversaire avant que tous les siens ne le soient.

Détails

- Chaque joueur possède une grille de jeu de 10 cases par 10 cases ainsi que 5 navires de longueur différentes :
 - Porte-avion (5 cases)
 - Croiseur (4 cases)
 - Contre-torpille (3 cases)
 - Sous-Marin (3 cases)
 - Torpilleur (2 cases)
- Avant de débiter, chaque joueur positionne secrètement ses 5 navires sur la grille de jeu.
- Dès lors, la partie peut débiter. Le premier joueur débute la partie en annonçant la case adverse visée par son tir (A1 par exemple).
- Ainsi, à tour de rôle, les joueurs annoncent une case où un tir est exécuté. Le joueur adverse annonce le résultat du tir selon ces trois possibilités :
 - tir manqué (le tir n'a atteint aucun navire)
 - tir réussi (le tir a atteint un navire sans toutefois le couler)
 - navire coulé (le tir a atteint la dernière case du navire)
- La partie termine dès qu'un joueur a perdu tous ses navires.

Précision

- Pour cette version Excel du jeu, chaque joueur doit posséder son propre fichier. Ainsi, deux joueurs peuvent jouer s'ils sont sur un réseau dont les dossiers sont partagés.
- Chaque joueur doit identifier les paramètres essentiels sur la feuille **Paramètres de combat** :
 - quel est le nom de son armée
 - qui est l'ennemi en spécifiant le nom du fichier de l'ennemi
 - qui est le joueur qui débute
- Ensuite, toute la partie se passe sur la feuille **Bataille navale**.
 - d'abord, chaque joueur doit positionner ses 5 navires en insérant les lettres 1 à 5 sur la grille de jeu
 - à tour de rôle, chaque joueur attend son tour et insère un x dans la grille de combat
 - le suivi se fait par des messages sur la zone de message ainsi que des codes de couleur à même la grille de jeu

Feuille 1 – Présentation

Paramètres du combat		Paramètres de sécurité masqués
Nom de l'unité de combat	Mon équipe	
Coordonnées de l'ennemi	Fichier ennemi.xlsx	
Joueur débutant les hostilités	Moi	

Feuille 2 – Paramètres de combat

Battleship

Mon équipe

VS

-



Code de construction
Longueur du bâtiment

Porte-avions	5	1
Croiseur	4	2
Contre-torpilleurs	3	3
Sous-marin	3	4
Torpilleur	2	5

COMMUNICATIONS



---#UT26025(T---

---+GP1*175,F---

Votre flotte est en construction.

Recherche de l'ennemi

Recherche de l'ennemi en cours

Veillez vous assurer que la cible est bien verrouillée.

Légende

Eau		Tir manqué	
Navires construits	1	Tir réussi	
Inconsistance	2	Navire coulé	

Légende affichée

Construction de sa flotte navale

	1	2	3	4	5	6	7	8	9	10
A										
B										
C										
D										
E										
F										
G										
H										
I										
J										

Suivi des tirs ennemis

	1	2	3	4	5	6	7	8	9	10
A										
B										
C										
D										
E										
F										
G										
H										
I										
J										

Message provenant de -

Suivi de ses propres tirs

	1	2	3	4	5	6	7	8	9	10
A										
B										
C										
D										
E										
F										
G										
H										
I										
J										

Message envoyé à l'ennemi

Planification de la construction en cours
Planification de la construction en cours
Planification de la construction en cours
Planification de la construction en cours
Planification de la construction en cours

-

-

-

-

-

-

Tirs totaux - Tirs manqués - Tirs réussis - Bâteaux coulés

-

-

-

-

-

-

Tirs totaux - Tirs manqués - Tirs réussis - Bâteaux coulés

Feuille 3 – Bataille navale