

Minis-projets - INSI

Année 2022-2023

Les mini-projets sont l'occasion de réaliser un premier projet en semi-autonomie.

Modalités :

Vous devez constituer des groupes (2 ou 3 élèves) et indiquer au professeur le sujet que vous avez retenu au plus tard le 13 décembre. Attention : les sujets seront attribués au fur et à mesure de l'inscription des groupes sur le modèle premier arrivé-premier servi.

A partir du 13 décembre, 1 h par semaine sera dédiée aux mini-projets. Cette heure doit servir essentiellement à vous concerter car le gros du travail doit être fait à la maison.

Chaque semaine le groupe passera une revue de projet de 5 minutes avec le professeur où il devra remettre sa fiche de suivi du projet (voir en fin de document) et où un point rapide sera fait.

Les projets doivent être terminés pour la mi-janvier. Il faudra alors réaliser un petit dossier pour présenter votre travail et vous passerez un oral collectif de 10-15 minutes fin janvier/ début février (le planning de passage sera donné ultérieurement).

Chaque élève devra remettre par voie électronique (messagerie de l'ENT) 3 jours avant son passage au plus tard, son programme complet (ensemble des fichiers sources) ainsi que son dossier (voir « dossier mini-projet » plus loin).

Evaluation :

Les mini-projets sont notés et cette note comptera pour le deuxième trimestre. Les critères d'évaluation sont donnés à la fin de ce document.

Sujets :

Les élèves sont libres de proposer un sujet de mini-projet (la proposition doit être suffisamment détaillée) qui pourra être accepté tel-quel, amendé ou refusé. Ils peuvent également choisir un sujet parmi la liste suivante (la difficulté est indiquée par le nombre d'étoiles ☆ et le nombre d'élèves dans le groupe par les petits personnages 👤) :

Sujets et descriptif succinct (les étoiles représentent la difficulté)	Noms
<u>Mastermind</u> ☆☆ (👤👤 - 👤👤👤) Réaliser en mode texte ou en mode graphique un jeu de Mastermind. Le joueur doit trouver la combinaison choisie par l'ordinateur. Celui-ci lui indique les pions de la bonne couleur mais mal placés et ceux de la bonne couleur et bien placés et compte le nombre d'essais pour trouver la solution.	
<u>Formatage de texte</u> ☆☆ (👤👤) Réaliser un programme capable de réaliser certains traitements sur un fichier texte comme : compter le nombre de phrases, de mots et de caractères, supprimer les accents, rajouter un saut de ligne à chaque phrase, mettre une majuscule à chaque début de phrase, limiter la largeur du texte à 80 caractères, ... et sauvegarder le résultat du traitement dans un fichier.	
<u>Jeu de Memory</u> ☆☆ (👤👤 - 👤👤👤) Réaliser un jeu de memory en mode texte dans un premier temps, puis en mode graphique. Le programme doit afficher le nombre de coups et le temps mis pour gagner. Voir http://www.jeu-test-ma-memoire.com/jeux-de-memory	
<u>Convertisseur Shadock-binaire</u> ☆ (👤👤) Concevoir un programme permettant de convertir un nombre écrit en Shadock en décimal ou l'inverse. Voir https://www.dcode.fr/numeration-shadok	
<u>Mini traitement d'image</u> ☆☆ (👤👤 - 👤👤👤) Ce programme doit être capable de charger une image, l'afficher puis réaliser un certain nombre de traitements sur l'image (passer en noir et blanc, inverser, supprimer un ou deux canaux de couleurs, flou, détection de contours ...) et enfin sauvegarder le résultat. Il faudra faire des recherches afin d'utiliser des bibliothèques externes de manipulation d'images.	

<p><u>Jeu de la vie</u> ☆ (👤👤)</p> <p>Le programme doit demander la taille de la grille et proposer soit d'utiliser une configuration de départ prédéfinie, soit une configuration aléatoire avec une probabilité de vie de chaque cellule réglable. Le programme affiche ensuite les différentes itérations avec une pause paramétrable entre chaque. Voir https://fr.wikipedia.org/wiki/Jeu_de_la_vie</p>	
<p><u>Fractales</u> ☆☆ (👤👤)</p> <p>Le programme doit demander la zone d'affichage (intervalle où tracer la fractale), puis proposer quelques exemples de fractales simples. Il trace ensuite la fractale et propose de sauvegarder le résultat dans un fichier image.</p> <p>Voir http://www.syti.net/Fractals.html , http://framy.free.fr/travaux_mathematiques.htm ou http://villemin.gerard.free.fr/Wwwgvmm/Suite/FracComp.htm#formule</p>	
<p><u>Logiciel de réservation</u> ☆ (👤👤)</p> <p>Un gîte souhaite informatiser son système de réservation. Pour cela il faudrait un logiciel qui demande à l'utilisateur de saisir sa date de réservation. Le programme devrait lire un fichier où sont situées les dates déjà retenues. Indiquer à l'utilisateur si la date est possible. Dans ce cas celle-ci sera à ajouter au fichier des dates sinon on demandera une nouvelle date.</p>	
<p><u>Logiciel d'entraînement aux opérations en ligne</u> ☆ (👤👤 - 👤👤👤)</p> <p>Une enseignante en CE2 souhaite réaliser un logiciel qui permette aux élèves de travailler en autonomie sur les opérations en ligne. L'utilisateur sera invité à choisir le type d'opération (addition, soustraction ou multiplication) et le programme devra générer une séquence de 10 opérations. A la fin le programme indiquera le nombre de bonnes réponses.</p> <p><i>Pour les additions et soustractions le nombre maximum de chiffres sera de 2, pour les multiplications 1 seul chiffre. De plus dans les soustractions le résultat doit être positif.</i></p>	
<p><u>Jeu du pendu</u> ☆☆ (👤👤 - 👤👤👤)</p> <p>On souhaite réaliser un "pendu". On suivra les règles classiques du jeu avec 10 erreurs maximum avant d'être pendu. On utilisera une bibliothèque restreinte de 10 mots à trouver dans le programme. Si l'on trouve le mot entier sans dépasser le nombre maximal d'erreurs, on affichera un message de victoire sinon perdu.</p>	
<p><u>Jeu des allumettes</u> ☆ (👤👤)</p> <p>On souhaite réaliser le « petit jeu des allumettes » (jeu présent à l'émission Fort Boyard). Ce jeu nécessite deux joueurs et 21 allumettes. Les 21 allumettes sont mises sur une même ligne (un même tas). Chacun à son tour, les joueurs piochent 1, 2 ou 3 allumettes dans le tas. Le joueur qui prend la dernière allumette perd la partie.</p> <p><i>Le programme devra pouvoir faire jouer au choix des joueurs humains ou artificiels. Le joueur artificiel pourra utiliser diverses stratégies.</i></p>	
<p><u>Bataille Navale</u> ☆☆☆ (👤👤👤)</p> <p>Jeu de bataille navale classique à 1 ou 2 joueurs. Les joueurs doivent pouvoir choisir la position de leurs bateaux. L'affichage de la grille peut se faire en mode texte ou en mode graphique. Le programme doit détecter la fin de partie et désigner le vainqueur.</p>	
<p><u>Arbitre de jeu d'échec</u> ☆☆ à ☆☆☆ (👤👤 - 👤👤👤)</p> <p>Deux joueurs jouent et l'ordinateur renvoie si le coup est valable, s'il y a échec, et qui a gagné. En mode texte ou avec une interface graphique.</p>	
<p><u>Démineur</u> ☆☆ (👤👤 - 👤👤👤)</p> <p>Jeu de démineur avec ou sans interface graphique.</p> <p>Voir : https://fr.wikipedia.org/wiki/D%C3%A9mineur_(genre_de_jeu_vid%C3%A9o)</p>	
<p><u>Mots de passe</u> ☆☆ (👤👤 - 👤👤👤)</p> <p>Programme avec plusieurs fonctions autour des mots de passes : s'assurer qu'un mot de passe proposé respecte certains critères de complexité (longueurs, caractères utilisés, ...) ; générer un mot de passe sécurisé aléatoire de longueur réglable ; casseur de mot de passe (trouve le mot de passe utilisé). Pour la dernière fonction une petite recherche sur la façon dont les mots de passe sont stockés sur un ordinateur sera nécessaire.</p>	
<p><u>Questions pour un champion</u> ☆ à ☆☆☆ (👤👤 - 👤👤👤)</p> <p>Jeu de questions à réponse libre. Le programme doit gérer une base de questions et en poser plusieurs successives au joueur humain. Il attribue des points si la réponse est juste.</p> <p>Il faudra définir les critères de fin de partie et éventuellement accepter des réponses proches ou synonymes.</p>	

Fiche de suivi de projet

Intitulé du projet :

Date :

Répartition du travail :

	Elève 1 :	Elève 2 :	Elève 3 :
Répartition des tâches :			
Tâches réalisées cette semaine			
Problèmes rencontrés			
Tâches à réaliser la semaine prochaine			

Notes :

Dossier mini-projet

Le dossier sur le mini-projet devra comporter 5 à 7 pages il est personnel (un dossier par élève) et être structuré comme suit :

- Couverture : Mini-projet : « Titre du projet », année, noms des participants (mettre en évidence le sien).
- *Présentation du programme et cahier des charges.*
- *Charte d'écriture du programme (optionnel pour les mini-projets).*
- *Répartition des tâches dans le groupe.*
- Présentation du déroulement du projet : phases successives de mise en œuvre, problèmes rencontrés et solutions apportées. Doivent apparaître dans cette partie la démarche de projet qui a conduit au résultat tel que présenté et la dimension collaborative du projet liée au travail en équipe.
- *Description des variables globales et des principales fonctions.*
- Description détaillée de la partie personnelle. Figurera en particulier le listing commenté du programme et si possible un algorithme¹.
- Conclusion personnelle sur le programme et les pistes d'améliorations possibles.
- Conclusion générale personnelle sur le mini-projet.

Les points en italique peuvent être les mêmes pour les différents membre d'un groupe de projet. Les autres éléments doivent être personnel.

¹ Voir https://fr.wikipedia.org/wiki/Organigramme_de_programmation et <http://troumad.developpez.com/C/algorigrammes/>

Oral mini-projet

L'oral du mini-projet se décompose en deux parties :

- Une présentation orale collective du travail par les candidats qui doit durer environ 5 minutes par personnes (10 minutes pour les groupes de deux et 15 minutes pour les groupes de trois). Il faut s'approcher le plus possible de cette durée sans toutefois la dépasser.
- Une séance de questions posées par le professeur et de conseils de 5 à 10 minutes.

Première partie : Présentation orale du travail

Dans cette partie, les candidats ne sont pas interrompus par l'examineur, sauf si la durée maximale est écoulée, auquel cas les candidats seront priés de conclure en 30 secondes.

La présentation s'appuie sur un diaporama (qu'il n'est pas nécessaire de fournir à l'avance mais qu'il faut avoir sur au moins 2-3 supports différents (clé USB, ENT, mail, ...)) et doit reprendre les éléments suivants :

- Présentation des membres du projet
- Sommaire de la présentation
- Présentation du projet (finalité, cahier des charges, ...)
- Organisation du projet (structure générale du programme, répartition des tâches, méthodes de travail employées (en particulier expliquer la façon dont vous avez coopéré), ...)
- Parties personnelles (Une ou deux pages par élève max. Détailler quelques algorithmes que vous avez programmé, justifier le choix d'une structure de donnée ou d'une technique utilisée, ...)
- Conclusion (votre sentiment sur le déroulement du projet, ce qu'il vous a apporté et les évolutions que l'on pourrait envisager, ...)

Deuxième partie : Dialogue avec le professeur

Cette partie dure le temps restant (minimum 5 minutes). Le professeur pose des questions aux candidats (soit à l'ensemble, soit à un en particulier) qui peuvent porter sur le projet en général et être plus précises quand elles touchent à la partie personnelle de l'élève. Elles peuvent s'appuyer sur ce qui a été mis dans le dossier ou sur la présentation qui vient d'être faite. Le professeur peut ensuite donner des conseils aux élèves pour s'améliorer lors du prochain projet.

Mini-projets - Evaluation

Nom :

Date :

Critères	Points
TRAVAIL	
Respect des délais impartis	/1
Qualité du suivi, de la gestion de projet (organisation, fiches suivi...)	/1
Quantité de travail	/1
	/3
PROGRAMME	
Fonctionnalité (le programme fonctionne-t-il ?)	/2
Lisibilité du code (indentations, sauts de lignes, noms des variables ...)	/0,5
Commentaires dans le code	/1
Qualité et efficacité du code	/1
Niveau de fonctionnalité atteint	/1
	/5,5
DOSSIER	
Respect des consignes et structure	/0,5
Qualité de l'analyse	/2
Qualité de la rédaction	/1,5
	/4
PRESENTATION ORALE	
Qualité du support de présentation (diaporama)	/1
Qualité de l'oral (élocution, utilisation de notes, dynamisme, ...)	/1
Qualité des explications	/1,5
Qualité des réponses lors de l'entretien	/1
Respect du temps imparti	/1
	/5,5
Globalisation	/2
TOTAL	/20