|  |
| --- |
| Le cnam |
|  |
| Rapport de Projet 2021 |
|  |
| Diviser-pour-Régner pour le calcul de l’enveloppe convexe |
| **Nom et Prénom** |
| **01/09/2021** |
|  |
|  |
|  |

Table des matières

[I. Introduction 3](#_Toc105791767)

[II. Partie I Ocaml 3](#_Toc105791768)

[III. Partie II JavaScript 3](#_Toc105791769)

[IV. Explications, difficultés et solutions 4](#_Toc105791770)

[IV. 1. Explications 4](#_Toc105791771)

[a) Ocaml 4](#_Toc105791772)

[b) Java Script 4](#_Toc105791773)

[IV. 2. Solutions 4](#_Toc105791774)

[a) Ocaml 4](#_Toc105791775)

[b) Java Script 4](#_Toc105791776)

[V. Conclusion 4](#_Toc105791777)

# Introduction

Je vais vous expliquer dans ce dossier comment j’ai répondue au cahier des charges pour les 2 applications qui ont été demander.

# Partie I Ocaml

En se qui concerne la partie de programmation en Ocaml, le sujet n’a été traité qu’en partie.

Malheureusement je n’ai réussi a créé le programme dans le temps imparti.

J’ai eu beaucoup de difficulté a trouver des ressources pour ce programme. En effet au premier abord, je n'ai rien trouvé dans les cours qui me permettaient de répondre aux demandes du programme. J'ai également longuement cherché sur Internet des renseignements sur l’algorithmes de Huffman. Sur Internet, j'ai fini par trouver des renseignements sur cet algorithme mais aucun ne correspondait ce que nous avions. Enfin, j'ai cherché sur Internet, plus largement des algorithmes pour lire un arbre binaire mais là aussi, je n'ai rien trouvé d'intéressant.

C'est seulement quelques jours avant la date de rendu que j'ai fini par trouver dans le court un algorithme qui se rapprochait de ce dont j'avais besoin. Je l'ai donc réadapté pour qu'il convienne a notre programme. Seulement je n'ai pas eu le temps d'aller plus loin.

C'est pour cela que mon programme n'est pas fonctionnel. Toutefois celui-ci contient les commentaires de toute ce que j'aurais voulu faire pour rendre ce programme fonctionnel.

Dans le programme Vous retrouverez un morceau de code qui permet de vérifier dans un arbre binaire si celui-ci contient ou non le paramètre fourni à la fonction (ex : 2, ‘h’, ‘B’).

En ce qui concerne les fonctions elles sont détaillées plus précisément en commentaire dans le programme

# Partie II JavaScript

En ce qui concerne le Java script j'ai également rencontré des difficultés. Toutefois elles ont été moindres car j'ai trouvé beaucoup plus de ressources plus facilement pour programmer ce petit jeu. En effet les premiers temps j'ai eu beaucoup de mal à comprendre la mécanique d'affichage en javascript avec le Canvas toutefois beaucoup de ressources étaient disponibles j'ai donc rapidement pu évoluer mon jeu.

le jeu contrairement à la programmation en Ocaml et fonctionnel je n'ai pas rencontré de bug particulier car je les ai corrigées au fur et à mesure qu'ils apparaissaient.

Toutes les fonctions ont été traitées.

Comme le programme en Ocaml le Js est commenté, chaque fonction est expliquée. J'ai également réussi à ajouter quelques fonctions bonus comme la difficulté qui augmente en fonction du nombre de points (de 0 à 1000 et de 1000 à 5000). En ce qui concerne le bouton qui affiche plusieurs vagues de soucoupes, je l’ai automatisé de manière à ce que son fonctionnement soit similaire à un interrupteur (On / Off).

# Explications, difficultés et solutions

## Explications

### Ocaml

Difficultés rencontrées en ocaml.

### Java Script

En Java script peu de difficultés rencontrées, toutefois quelques petits bugs ont été corrigés.

Notamment quand l'utilisateur clique sur le bouton pour faire apparaître une seul soucoupes. Une fois qu'il retourne dans le jeux et tirer un missile (avec la barre espace), le bouton est pressé à nouveau ce qui crée des soucoupes (en boucle) alors que ce n'est pas ce qui est voulu.

## Solutions

### Ocaml

Problème solutionné en ocaml grâce à de la recherche sur internet et aux révisions des exercices fournis en cours.

### Java Script

Pour résoudre le problème du bouton pour qui est « appuyé » à chaque nouveau tir, j'ai utilisé une petite fonction en javascript. Cette fonction a été ajoutée aux 2 boutons et me permet de remettre le focus sur le canvas afin que le bouton ne soit plus sélectionné et donc plus concerné par la barre espace quand le joueur effectué un tir.

# Conclusion

En conclusion je dirais que je ne suis qu’à moitié satisfait de mon rendu car je n'ai pas eu le temps de m'occuper de la partie en Ocaml, toutefois je suis très fier de mon jeu en Java script car il est très abouti et je n'ai pas rencontré de bug.