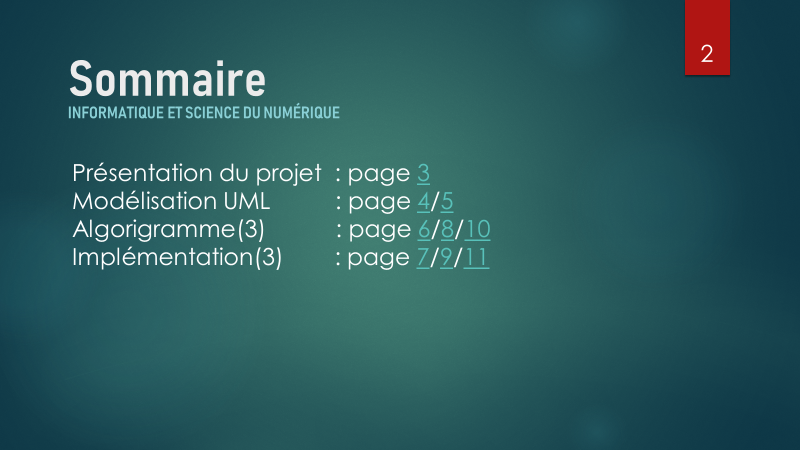
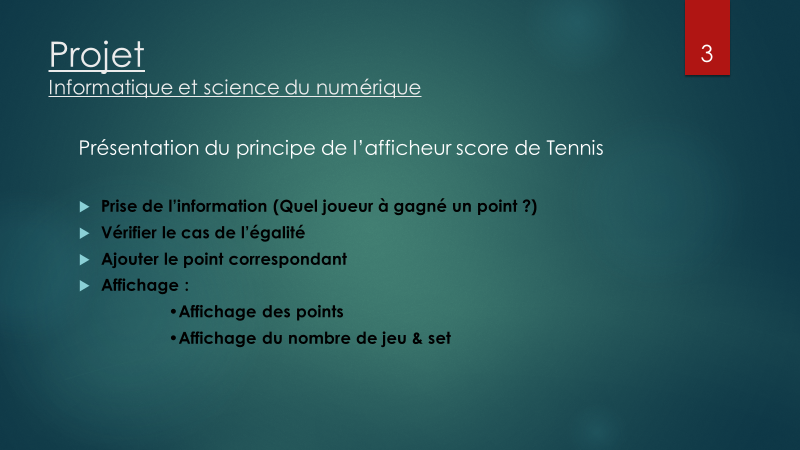
DOSSIER EPREUVE ISN

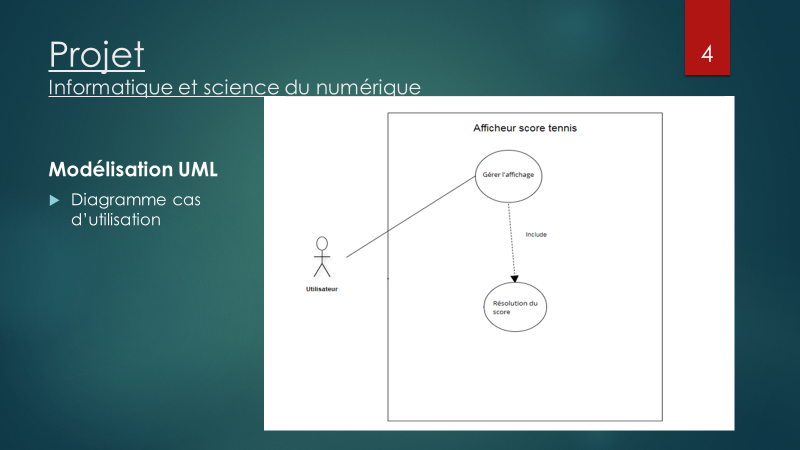
ICARD Sébastien



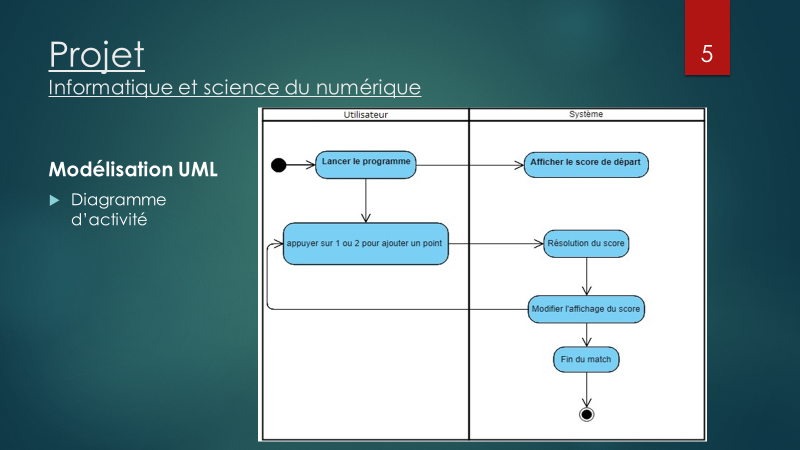




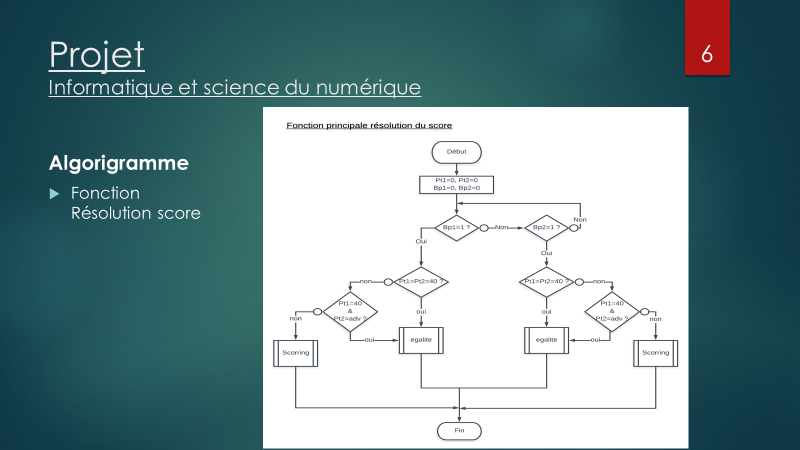
Le projet consiste à facilité le comptage des points de tennis pour un arbitre. L’arbitre n’a plus qu’à appuyer sur la touche 1 ou 2 correspondante au joueur qui a marqué un point.



L’utilisateur agis sur la fonction Affichage qui elle-même agis sur la fonction résolution du score.



Modélisation UML sous forme de diagramme d’activité du projet dans sa globalité.



Algorigramme correspondant à la fonction principale de mon programme.

**Implémentation de cette fonction (resolution\_du\_score) :**

#include <stdio.h>

#include <stdlib.h>

//Déclaration des fonctions

int resolution\_du\_score(int joueur\_gagnant, int \*pointeur\_score\_player\_1, int \*pointeur\_score\_player\_2,int \*pointeur\_test);

int get\_next\_score(int point);

int get\_egal\_score(int point);

//fonction traitant du scoring

int resolution\_du\_score(int joueur\_gagnant, int \*pointeur\_score\_player\_1, int \*pointeur\_score\_player\_2,int \*pointeur\_test)

{

//on test si l'arbitre a appuyé sur 1 ou 2

if (joueur\_gagnant=='1'|| joueur\_gagnant=='2')

{

// si on appuie sur la touche 1 ajouter un point au joueur 1

if (joueur\_gagnant=='1')

{ // on test si il y a egalité (40-40)

if(\*pointeur\_score\_player\_1==\*pointeur\_score\_player\_2 && \*pointeur\_score\_player\_1==40)

{

\*pointeur\_score\_player\_1 = get\_egal\_score(\*pointeur\_score\_player\_1);

}

//on test si il y a avantage

else if(\*pointeur\_score\_player\_1==40 && \*pointeur\_score\_player\_2==50 )

{

\*pointeur\_score\_player\_2 = get\_egal\_score(\*pointeur\_score\_player\_2);

}

//sinon on ajoute un point classique au joueur 1

else

{

\*pointeur\_score\_player\_1 = get\_next\_score(\*pointeur\_score\_player\_1);

}

}

// si on appuie sur la touche 2 ajouter un point au joueur 2

if (joueur\_gagnant=='2')

{ // on test si il y a egalité (40-40)

if(\*pointeur\_score\_player\_1==\*pointeur\_score\_player\_2 && \*pointeur\_score\_player\_1==40)

{

\*pointeur\_score\_player\_2 = get\_egal\_score(\*pointeur\_score\_player\_2);

}

//on test si il y a avantage

else if(\*pointeur\_score\_player\_2==40 && \*pointeur\_score\_player\_1==50 )

{

\*pointeur\_score\_player\_1 = get\_egal\_score(\*pointeur\_score\_player\_1);

}

//sinon on ajoute un point classique au joueur 2

else

{

\*pointeur\_score\_player\_2 = get\_next\_score(\*pointeur\_score\_player\_2);

}

}

return 0;

//Ici on informe que tout c'est bien passé

\*pointeur\_test=0;

}

/\*Parcontre si l'arbitre a appuyé sur une autre touche que 1 ou 2

on renvoie un message d'erreur et on ne traite pas le score \*/

else

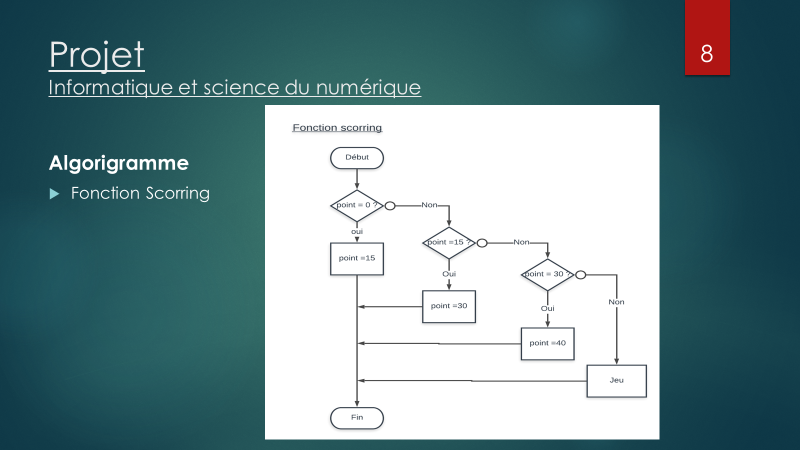
{

\*pointeur\_test=1;

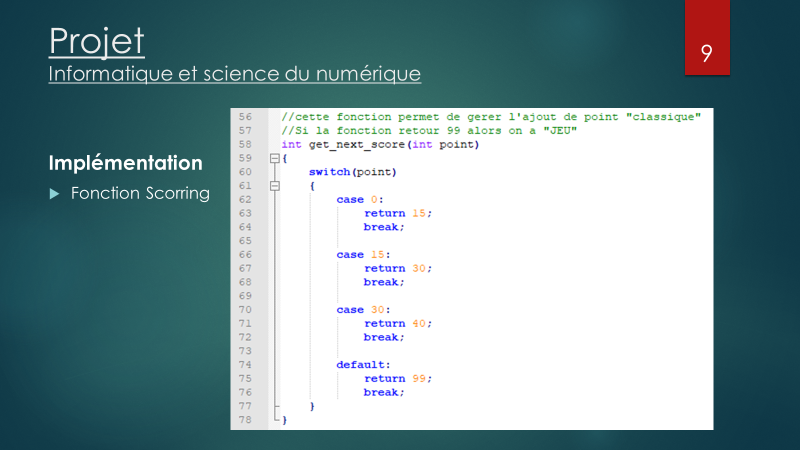
}

}

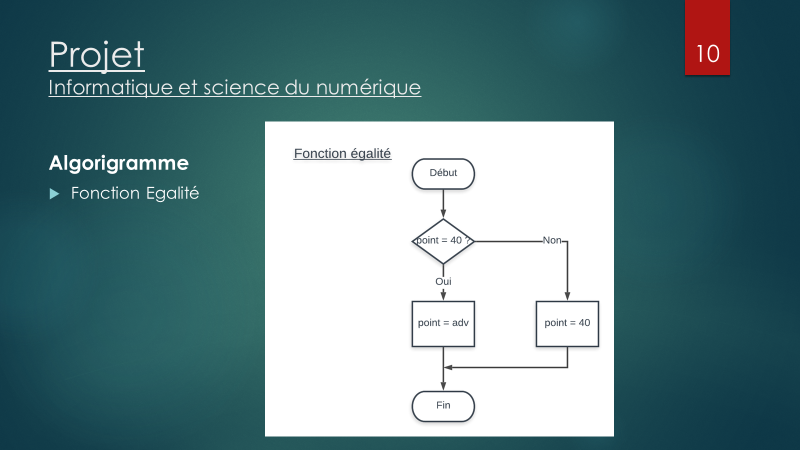
**Pour plus de clarté** [**cliquez ici**](../3.%20Code%20source/scoring.c)



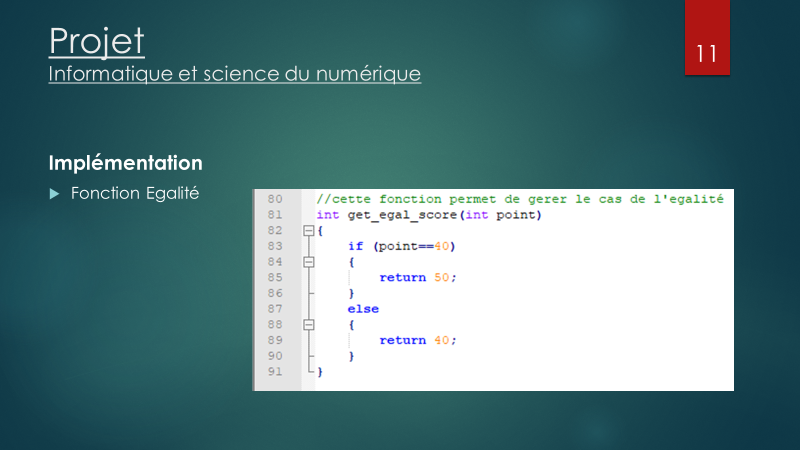
Algorigramme traitant de la fonction Scorring qui est appelé dans la fonction principal(resolution\_du\_score)



Implémentation de cette fonction « scoring »  ( ici appelé get\_next\_score )



Algorigramme traitant de la deuxième fonction appelé dans la fonction principal(resolution\_du\_score)



Implémentation de cette fonction « Egalité » (appeé ici get\_egal\_score)

**CLASSE DE TEST :**

Ici nous allons tester notre programme. Nous lui enverrons donc de multiple valeur(aléatoire) afin de voir s’il y’a une erreur, ces tests sont concluant avec 100% de réussite on peut donc affirmer que notre programme pourra fonctionner sans problème dans la plupart des cas :

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <time.h>**

**#include "Scoring.c"**

**const int max\_size = 5;**

**const int taille =103;**

**const char set[] = "azertyuiopqsdfghjklmwxcvbnAZERTYUIOPQSDFGHJKLMWXCVBN1234567890&""'(-\_)=$\*!:;,?./§µ%£¨+°<>}]@^\\`|[{#~²"; // 103 caractères**

**int test=0;**

**int main()**

**{**

**int b=0;**

**int i = 0;**

**int score\_player\_2=0;**

**int score\_player\_1=0;**

**srand(time(NULL)); // initialisation de rand**

**FILE\* fichier = NULL;**

**char joueur\_gagnant[max\_size];**

**time(NULL);**

**FILE \*fp = fopen("classe\_de\_test\_char.txt", "w+");**

**if ( fp == NULL )**

**{**

**fputs("Le fichier texte.txt n'a pas pu être ouvert\n", stderr);**

**return -1;**

**}**

**for(b=0; b<100000; b++)**

**{**

**for (int j = 0; j < 5; j++ )**

**{**

**i = rand()%taille;**

**joueur\_gagnant[j] = set[i];**

**resolution\_du\_score(joueur\_gagnant, &score\_player\_1, &score\_player\_2, &test);**

**}**

**if (joueur\_gagnant=='1' || joueur\_gagnant=='2' && test==0)**

**{**

**fprintf(fp,"PASSED\n");**

**}**

**if (joueur\_gagnant!='1' || joueur\_gagnant!='2' && test==1)**

**{**

**fprintf(fp,"PASSED\n");**

**}**

**else**

**{**

**fprintf(fp,"FAILED\n");**

**}**

**}**

**printf("fin de la classe de test");**

**fclose(fp);**

**return 0;**

**}**

**Pour plus de clarté** [**cliquez ici**](../5.%20classe%20de%20test/class_de_test_char/main.c)

**Conclusion :**

Pour conclure, nous avons pu mener à bien ce projet en ce répartissant les tâches, à savoir : - Affichage (Antonin Silvestre)

-Résolution du score (moi)

Cella nous a appris à gérer notre temps, gérer les erreurs et bugs en y apportant les corrections nécessaires.

ICARD SEBASTIEN TS3