Rapport du projet de programmation

Sujet : jeu de cartes belote

Réalisé par :

Firas Jaadari

Issam Hedhli

Issa Taleb

sous la direction de: Mme Imen Boukhris

AU : 2020/2021

ENSI
ÉCOLE NATIONALE DES SCIENCES DE L'INFORMATIQUE

Remerciements

Nous souhaitons adresser les remerciements les plus sincères aux personnes qui ont nous ont apporté leur aide et qui ont contribué à l'élaboration de ce projet spécialement Mme Imen Boukhris pour ses encouragements et son encadrement durant toute cette période

Sommaire :

INTRODUCTION :

Le but de ce projet consiste à programmer une version simplifié du Jeu de Belote.

Après avoir appris la programmation objet orienté, ce projet devrait nous faire pratiquer l'objet en C++.

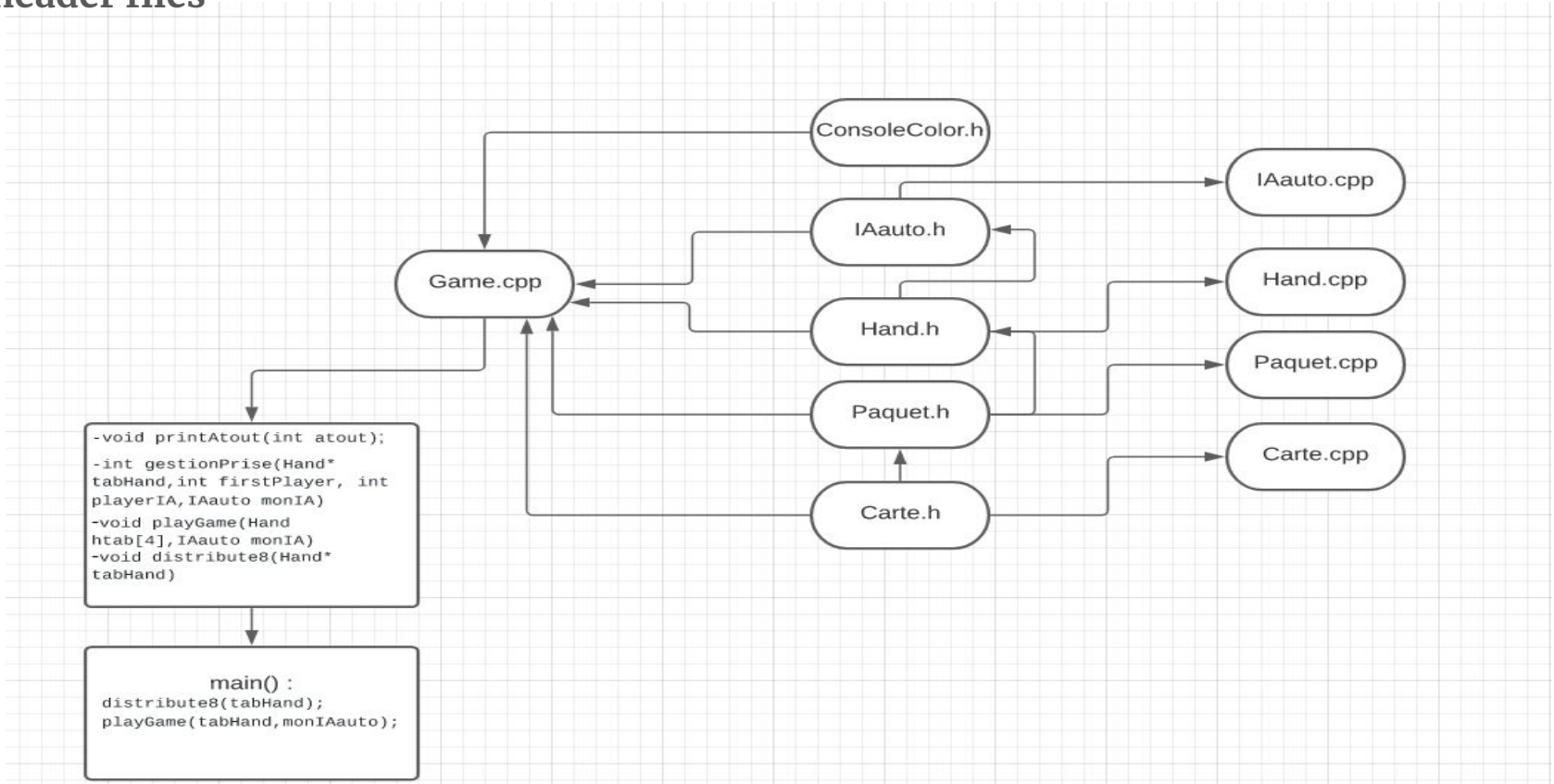
Nous avons détaillés les caractéristiques du jeu et les règles dans les parties suivantes.

présentation du jeu de belote

LE JEU :
C’ est un jeu de cartes qui se joue avec un paquet de 32 cartes comprenant 4 bois (Pique, Cœur, Carreau et Trèfle) et 8 valeurs.
trois de ces quatre joueurs sont des joueurs humains, l’autre joueur se fait joué par l’ordinateur.
Les joueurs forment deux équipes de deux. Les joueurs 1 et 3 forment première équipe, et les joueurs 2 et 4 forment deuxième équipe.
A la fin de la partie, chaque équipe compte ses points en additionnant les points des cartes qu’elle a gagnées.

Architecture du programme

Le programme principale est dans le fichier game.cpp qui appelle toutes les autre header files



CLASS Carte

MÉTHODES:

constructeur : Carte();
getValue())
geAtout()
getPoint()
getOrdre()
isSuperieur(Carte c)
setAtout()
surchage des operateur :
<< ; == ; <

ATTRIBUTS:

private :
int value,color,point
,ordre;
Bool isAtout

public :
enum color_type {COEUR,
PIQUECARREAU, TREFLE};
enum value_type {SEPT, HUIT, NEUF,
VALET, DAME, ROI, DIX, AS};

CLASS Paquet

MÉTHODES:

Constructeur : Paquet();

ATTRIBUTS:

int jeu[32];
int nbCarte ;

CLASS IAauto

METHODES :

int minimax(int firstPlayer,
int atout, Hand *);
int minimax(state c);
int maxValue(state c);
int minimaxAlphaBeta(int
firstPlayer, int atout, Hand
*);
int minimax(state c, int
alpha, int beta);

ATTRIBUTS:

int
player,partner,nbFin,cpt,cardsStillI
nGame[32] ,nbCardDeleted ,nbTour

Hand h;
struct state{
int nbPlayerPli,playerHasHand,
Playerwinningpli,int atout ,
askcolor nbCarte, valuePli,score;
, Carte Bestcard;
Hand currentHand , allHand[4];};

CLASS IAauto

METHODES :

int maxValue(state c, int alpha, int beta);

int minValue(state c, int alpha, int beta);

state majState(state c, int i);

void deleteCard(Carte c);

Carte nextCarte(Hand*, int atout, int player, int numberCardPlayedInPli, Carte bestCard, int colorAsk, int valuePli, int playerWining);

METHODES:

void distributionCards(Hand h, Hand[4], int atout);

void printGame(Hand htab[4]);

void delListCard(Carte c[], int lg);

bool isCarteValide(Hand h, Carte c, int colorAsk, int atout, Carte bestCard, int winner, int actualPartner);

CLASS IAauto

METHODES :

Carte carteAuto(Hand h,int colorAsk,int atout, Carte bestCard,int winner,int actualPlayer);

int prendre(Hand h,int firstPlayer, int playerIA);

int prendreScore(Hand h, int atout, int firstPlayer, int playerIA);

METHODES

void distributionPrise(Hand h, Hand* htab,int atout);

constructeur :IAauto();

IAauto(int nb);

~Destructeur : IAauto(void);

CLASS Hand

METHODES :

void deleteCarte(int index);

void deleteCarte(Carte c);

int posColor(int color);

bool hasColor(int c);

bool hasAtoutSup(int ordre, int atout);

int Hand::nbColor(int c);

void triAtout(int color);

void setAtout(int atout);

ATTRIBUTS:

int nbCarte,posPique,posCoeur,posCarreau,posTrefle, nbPique,nbCoeur, nbCarreau,nbTrefle;

Carte listHand[8];

CLASS Hand

MÉTHODES :

private :
void triABulle(Carte
tableau[], int longueur);
constructeur :
Hand();
Hand(Carte*, int);
destructeur :~Hand(void);

fonctionnement du programme

classe Hand : pour la gestion des cartes de chaque joueur
class Carte : pour la construction et l'utilisation des cartes
class IAauto : déroulement du jeu automatique pour l'ordinateur
Classe reprenant les 32 cartes d'un jeu
ConsoleColor.h : fichier headerfile pour colorer les sortie du console

les problèmes rencontrés :

comprendre et apprendre les concept du jeu

comprendre l'algorithme minimax alpha beta

les interfaces graphiques

les solutions

1-concept du jeu : demander aux amis , essayer de jouer des partie en direct et en ligne , regarder des tutos

2-Algorithme minimax alpha beta : c'est un algorithme prédéfinie pour les jeu multijoueurs , on a constaté des explications sur youtube et des exemples des codes pour comprendre comment implémenter cet algorithme

3-Les interfaces graphiques : on a testé plusieurs librairies et frameworks ,on a suivis le cours de Qt sur openclassrooms , on est entré dans une contrainte du temps on n'arrivait pas a terminer l'interface graphique .

Menu Principal:

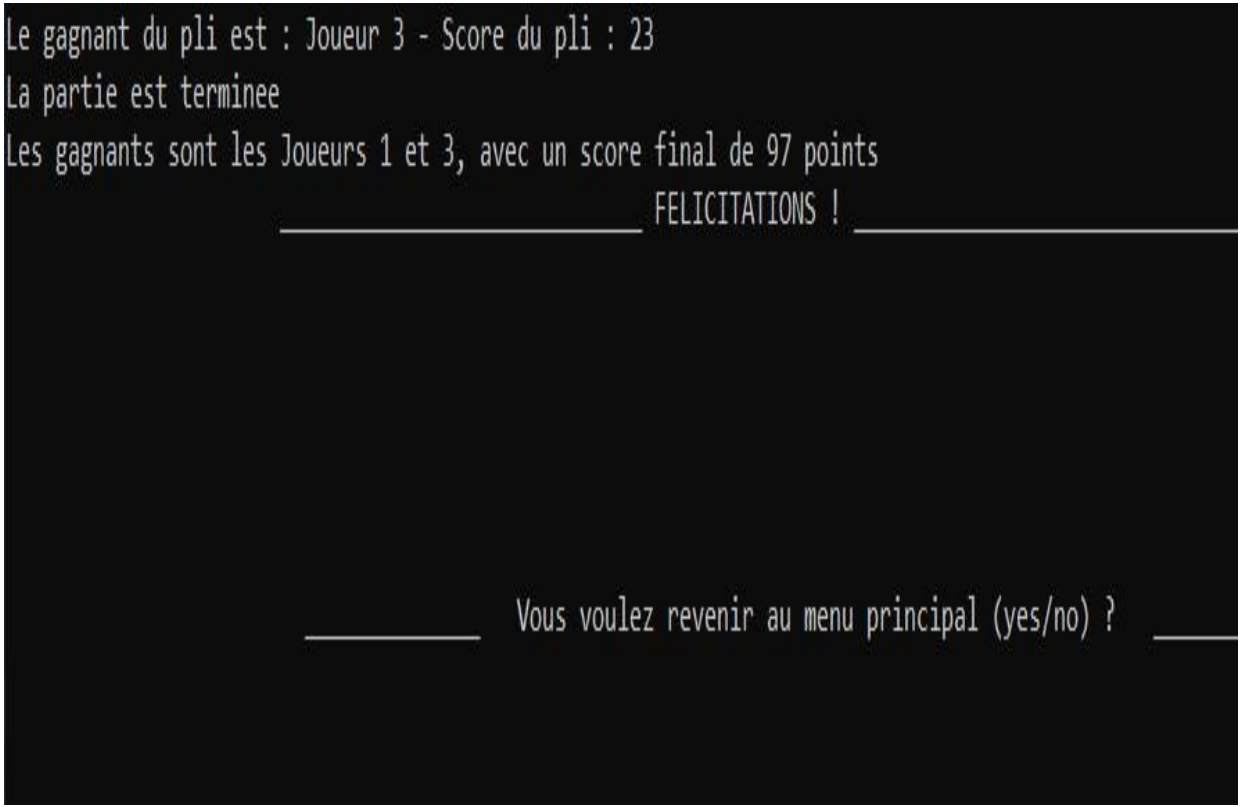
Début du jeu :

```
Belote : nouveau jeu
Choisir le joueur qui commence la partie (0 - 3) : 2
Atout par defaut : pique
--------------------------------------------------------------------------------
.... Tour 1 ....
--- Joueur 0 -- + --- Joueur 1 -- + --- Joueur 2 -- + --- Joueur 3 -- +
0 Dix   Coeur   | 0 Sept  Coeur   | 0 Huit  Coeur   | 0 Valet Coeur   |
1 Neuf  Coeur   | 1 As    Coeur   | 1 Roi   Coeur   | 1 Neuf  Pique   |
2 Valet Pique   | 2 Dame  Coeur   | 2 As    Pique   | 2 Dame  Pique   |
3 Roi   Pique   | 3 Dix   Pique   | 3 Dame  Carreau | 3 Huit  Carreau |
4 Huit  Pique   | 4 Sept  Pique   | 4 Roi   Trefle  | 4 Roi   Carreau |
5 Sept  Carreau | 5 Dix   Carreau | 5 As    Trefle  | 5 Neuf  Carreau |
6 Sept  Trefle  | 6 Valet Carreau | 6 Neuf  Trefle  | 6 As    Carreau |
7 Huit  Trefle  | 7 Dame  Trefle  | 7 Valet Trefle  | 7 Dix   Trefle  |
--------------------------------------------------------------------------------
.... Joueur 2, (1/4) ....
Joue quoi ?
```

déroulement du jeu :

```
.... Joueur 2, (1/4) ....
Joue quoi ? 1
La carte jouee est : Roi   Coeur
.... Joueur 3, (2/4) ....
Joue quoi ? 3
Carte non acceptee selon les regles du jeu !
Joue quoi ? 1
Carte non acceptee selon les regles du jeu !
Joue quoi ? 0
La carte jouee est : Valet Coeur
Valet Coeur   est superieur a Roi   Coeur    ? => 0
.... Joueur 0, (3/4) ....
La carte jouee est : Neuf  Coeur
Neuf  Coeur   est superieur a Roi   Coeur    ? => 0
.... Joueur 1, (4/4) ....
Joue quoi ? 0
La carte jouee est : Sept  Coeur
Sept  Coeur   est superieur a Roi   Coeur    ? => 0
Le gagnant du pli est : Joueur 2 - Score du pli : 6
```

fin du jeu:


Le gagnant du pli est : Joueur 3 - Score du pli : 23
La partie est terminee
Les gagnants sont les Joueurs 1 et 3, avec un score final de 97 points
FELICITATIONS !
Vous voulez revenir au menu principal (yes/no) ?

conclusion :

Nous avons créé des méthodes utiles pour un jeu belote réel dans les classes. En effet nous avions prévus d'avoir programmer le jeu belote réel. Par contre le temps pour rendre le projet était un peu court, alors nous ne sommes pas arrivés à le
faire
Comme nous avons vu un peu d'interface graphique, nous
avons eu le souhait de réaliser une interface graphique pour ce projet.
Ce qui n'a pas été le cas pour faute de temps.