ISN

*Projet Blackjack*



Jérémy TRULLIER, Soumaya BENHADHOU, Natacha PADOVA

## Présentation :

Qu’est-ce que le blackjack ?

Le blackjack a fait son apparition à la fin du XVIIIe siècle sous le nom de « 21 ». Il est à l’époque joué dans les casinos parisiens. Il est ensuite introduit aux États-Unis sans grand succès. Pour essayer d’attirer les joueurs, les casinotiers inventent des bonus. Les joueurs ayant un As de pique et un Valet noir gagnent un bonus qui offre alors un paiement de 10 contre 1. Le nom du jeu provient donc du « Valet Noir » de ce bonus soit « Black Jack » en anglais. On appelle encore ce jeu ainsi, bien que ce bonus ait été modifié. Ce jeu est encore l’un des plus populaires dans les casinos.

Chaque joueur qui se trouve autour de la table reçoit deux cartes, y compris le croupier. Celles du joueur sont faces découvertes, le croupier n'en montre lui qu'une seule. Que les autres joueurs puissent voir vos cartes ou pas n'a pas d'importance dans la mesure où vous ne jouez que contre la Maison.

Le but du jeu est de battre le croupier en accumulant un score de main plus élevé, sans dépasser 21.

Le score de votre main est la somme des différentes cartes. Les cartes avec un chiffre comptent pour leur valeur. Les cartes habillées valent 10 points alors qu'un As peut compter pour un ou pour onze points, au choix.

Pourquoi le Blackjack ?

Suite à un processus de réflexion, notre groupe ayant la volonté de réaliser un jeu pour notre projet de fin d’année, nous avons opté pour la réalisation d’un jeu de Blackjack. Nous désirions programmer un jeu nécessitant une interface graphique et le Blackjack s’y prêtait. De plus ce jeu apportait une dimension mathématique au programme avec le comptage des cartes qui fait appel à des calculs de probabilités.

Pyhton et ses modules :

Python est le langage de programmation que nous avons appris cette année en ISN. C’est un langage qui est assez simple d’accès avec sa simple syntaxe et qui propose tout de même de nombreux modules permettant de gérer différents paramètres, notamment l’image et l’aléatoire. Pour notre projet il nous a fallu utiliser le module random qui implémente un générateur de nombres pseudo-aléatoire pour différentes utilisations. Nous avons aussi eu recourt à Tkinter qui est un module de création d’interface graphique qui nous a permis de faciliter l’utilisation du jeu.

Notre code peut être divisé en 2 grandes parties : la première est le jeu sous forme textuelle qui permet de jouer au Blackjack en mode « texte » sur la console ; la deuxième est l’interface graphique qui illustre le jeu. Dans cette partie on crée une fenêtre Tkinter qui pourra afficher les cartes, les scores, et le reste du jeu.

Cahier des charges :

* Un programme « decklist » pour toutes les fonctions nécessaires au jeu : tirer des cartes, compter la valeur de la main, contrôler le croupier, déterminer le gagnant de chaque manche
* Une interface graphique  pour simplifier l’utilisation du jeu (affichage des cartes, scores, etc…)"
* Un programme « globalVars » pour gérer les variables globales qu’on utilise pour compter les scores et les probabilités de tirer une figure ou une petite carte.

Répartition des tâches :

|  |  |
| --- | --- |
| **Tâches** | **Réalisation** |
| plan du code | Jeremy |
| Deck + re-mélanger les cartes | Natacha |
| fonction tirer | Natacha |
| fonction total | Natacha |
| fonction croupier | Soumaya |
| fonction joueur | Soumaya |
| fonction gagnant | Soumaya |
| boucle principale | Jeremy |
| cadre de l'interface graphique | Jeremy |
| fonction afficher les cartes | Jeremy |
| fonction rejouer | Jeremy |
| fonction distribuer | Jeremy |
| boutons carte/rester | Jeremy |
| bouton rejouer | Jeremy |
| bouton aide | Soumaya |
| compter les cartes | Natacha |
| recherches sur le BJ | Tous |

Pour réaliser ce projet nous avons eu recourt à la plateforme GitHub, un logiciel de partage en ligne, qui nous a permis a mise en commun des recherches et du code, alors accessible à tout instant et pouvant être modifié.

Programme :

Notre programme peut être séparé en deux parties principales : la première est le jeu du blackjack qui permet de jouer au jeu en interaction « texte » et qui prend en compte tous les paramètres du jeu (il est lui-même composé de plusieurs sous-fonctions, chacune gérant un paramètre de jeu) ; la deuxième partie est l’interface graphique qui simplifie l’interaction du joueur avec le croupier.

Démarche :

Pour implémenter le jeu de Blackjack, d’abord sur la console python, puis sur Tkinter, nous avons fait une liste des fonctions à utiliser.

Il fallait d’abord créer un « deck » avec les cartes du jeu, en leur attribuant une valeur. Ensuite, une fonction qui tire des cartes et les ajoute à une main, une fonction qui calcule la valeur de la main.

Nous avons mis en place une fonction qui joue le croupier et une autre qui permet au joueur de prendre des décisions, et enfin, une fonction qui détermine le gagnant de chaque tour.

Pour l’interface graphique, nous avons commencé par faire des tests pour maitriser Tkinter. Pour finir, nous avons transcrit le jeu en mode « console » sur l’interface graphique :

Le programme agit d’après la logique de l’organigramme ci-dessous :

**Organigramme :**

Calculer la main

Comparer le score

Afficher

Perdu

Tirer 2 cartes pour le joueur

Tire une carte pour le croupier

Deck

Début

Fin

Rester

Quitter / rejouer ?

Quitter

Initialiser les données

Tirer / rester

Tirer

+ De 21 ?

Oui

Non

Calculer la main

Croupier tire

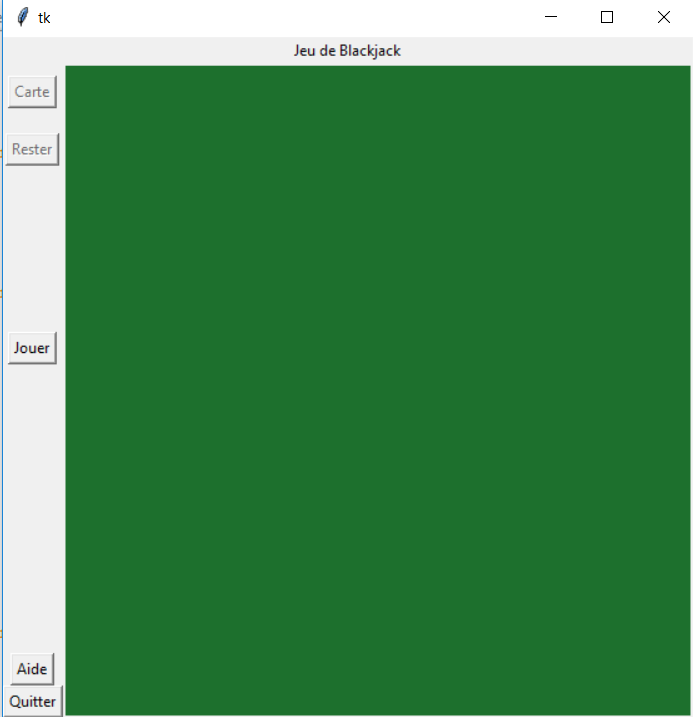
+ De 17?

Oui

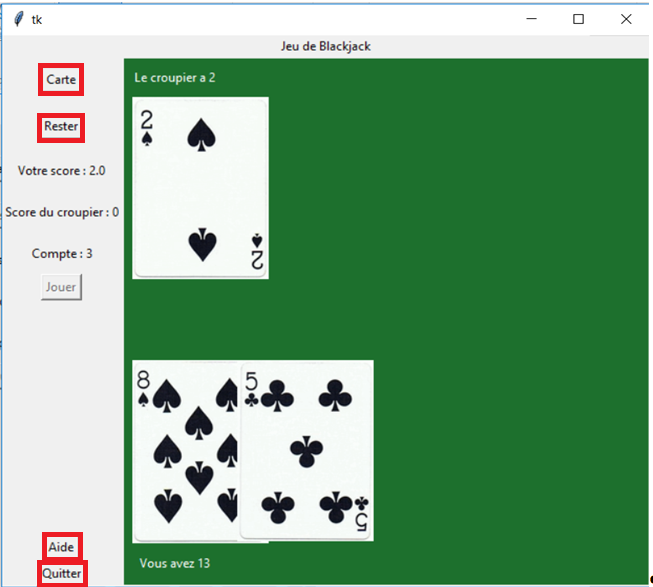
Rejouer

Exemple :

Lorsqu’on lance le programme cette fenêtre s’affiche

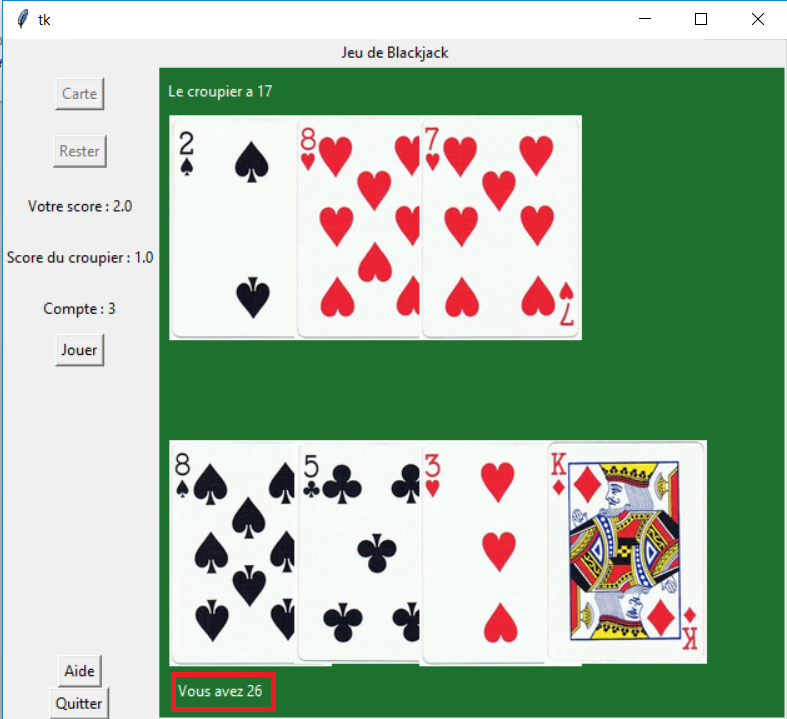


En cliquant sur le bouton jouer le jeu ainsi que l’affichage des cartes aléatoires va s’enclencher :



Et on peut ainsi choisir l’option de tirer une carte ou de rester sans tirer. On peut aussi demander de l’aide : une fenêtre avec le tableau de la stratégie de base s’ouvrira alors pour guider le joueur. Le bouton quitter ferme la fenêtre et interrompt la partie.

En suivant l’une des deux options de jeu, on peut arriver jusqu’à la fin du tour où le gagnant est alors décidé :



Ici le joueur a perdu face à la banque car la valeur de sa main est supérieure à 21.

## Code :

Le code est composé de 3 fichiers pythons :

* Le fichier « globalVars.py » :

# Variables globales

#Compte les cartes pour "prédire" s'il faut tirer ou pas

compte **=** 0

#Comptabilise le nombre de victoires du joueur et du croupier.

Scores **=** **[**0**,**0**]**

* Le programme « decklist.py » :

**import** globalVars

dictDECK **=** **{**'as\_coeur'**:**11**,**'2\_coeur'**:**2**,**'3\_coeur'**:**3**,**'4\_coeur'**:**4**,**'5\_coeur'**:**5**,**'6\_coeur'**:**6**,**'7\_coeur'**:**7**,**'8\_coeur'**:**8**,**'9\_coeur'**:**9**,**'10\_coeur'**:**10**,**'valet\_coeur'**:**10**,**'dame\_coeur'**:**10**,**'roi\_coeur'**:**10**,**'as\_carreau'**:**11**,**'2\_carreau'**:**2**,**'3\_carreau'**:**3**,**'4\_carreau'**:**4**,**'5\_carreau'**:**5**,**'6\_carreau'**:**6**,**'7\_carreau'**:**7**,**'8\_carreau'**:**8**,**'9\_carreau'**:**9**,**'10\_carreau'**:**10**,**'valet\_carreau'**:**10**,**'dame\_carreau'**:**10**,**'roi\_carreau'**:**10**,**'as\_trefle'**:**11**,**'2\_trefle'**:**2**,**'3\_trefle'**:**3**,**'4\_trefle'**:**4**,**'5\_trefle'**:**5**,**'6\_trefle'**:**6**,**'7\_trefle'**:**7**,**'8\_trefle'**:**8**,**'9\_trefle'**:**9**,**'10\_trefle'**:**10**,**'valet\_trefle'**:**10**,**'dame\_trefle'**:**10**,**'roi\_trefle'**:**10**,**'as\_pique'**:**11**,**'2\_pique'**:**2**,**'3\_pique'**:**3**,**'4\_pique'**:**4**,**'5\_pique'**:**5**,**'6\_pique'**:**6**,**'7\_pique'**:**7**,**'8\_pique'**:**8**,**'9\_pique'**:**9**,**'10\_pique'**:**10**,**'valet\_pique'**:**10**,**'dame\_pique'**:**10**,**'roi\_pique'**:**10**}**

# deck = ensemble de cartes du jeu

# dictDECK est le dictionnaire qui permet d'attribuer une valeur aux cartes

deck **=** list**(**dictDECK**)** #dictDECK est un dictionnaire

#il faut donc le transformer en liste pour ne garder que les cartes dans deck

**import** random

random**.**shuffle**(**deck**)** #Mélange le deck

#Fonction qui tire un nombre de cartes donné.

#En pratique, on tire une ou 2 cartes à la fois.

**def** **tirer(**hand**,**x**):**

**if** x **>** len**(**deck**):** #vérifier que le deck n'est pas vide

**print(**"Pas assez de cartes."**)**

recharge **=** list**(**dictDECK**)**

random**.**shuffle**(**recharge**)**# alors, mélanger la recharge

deck**.**extend**(**recharge**)**

globalVars**.**compte **=** 0

**for** i **in** range**(**x**):**

# ne pas utiliser '=', sinon, ne modifie pas la variable passée en paramètre

carte **=** deck**.**pop**(**0**)**

valeur **=** dictDECK**.**get**(**carte**)**#compte les cartes pour "prédire" s'il faut tirer ou pas

**if** 2 **<=** valeur **<=** 6**:**

globalVars**.**compte **+=** 1

**elif** valeur **>=** 10**:**

globalVars**.**compte **-=** 1

hand**.**append**(**carte**)**

#print("Le compte est à "+str(globalVars.compte))

#Fonction qui compte la valeur totale de la main

**def** **total(**hand**):**

t **=** 0

nbrAS **=** 0

nbrCartes **=** len**(**hand**)**

**for** i **in** range**(**nbrCartes**):**#prend la valeur de la carte dans dictDECK

carte **=** hand**[** i **]**

valeur **=** dictDECK**.**get**(**carte**)**

**if** valeur **==** 11**:**#Compte le nombre d'as dans la main

nbrAS **+=** 1

t **=** t **+** valeur

**while** nbrAS**>**0 **and** t**>**21**:**

t **-=** 10

nbrAS **-=** 1

**return** t

**def** **croupier(**hand\_croupier**):**

#Le croupier a déjà une ou plusieurs cartes

**while** total**(**hand\_croupier**)** **<** 17**:** #Le croupier tire jusqu'a 17 et s'arrete au dessus

tirer**(**hand\_croupier**,**1**)**

**print(**hand\_croupier**)**

total\_croupier **=** total**(**hand\_croupier**)**

**print(**"Score du croupier : " **+** str**(**total\_croupier**)** **)**

**def** **joueur(**hand\_joueur**):**

#le joueur choisit s'il tire des cartes ou s'il reste

**while** total**(**hand\_joueur**)** **<** 21**:**

x**=**input**(**"Voulez vous une autre carte ?\nSi oui, tapez h, sinon tapez sur Entrée:"**)**

**if** x **==** "h"**:**

tirer**(**hand\_joueur**,**1**)**

**print(**"Votre score : " **+** str**(**total**(**hand\_joueur**))** **)**

**print(**hand\_joueur**)**

**else:**

**break**

#Fonction qui compare le score de chaque joueur et affiche le nombre de victoires

**def** **gagnant(**x**,**y**,**s**):**

#x est le total de la main du joueur

#y est le total de la main du croupier

#s est une liste de scores dont le premier élément est le nombre de victoires du joueur et le second celui du croupier

**if** y **>** 21**:**

**if** x **>** 21**:**

**print(**"ÉGALITÉ"**)**

**return** 0

**else:**

**print(**"GAGNÉ"**)**

s**[**0**]** **+=** 0.5

**return** 0.5

**else:**

**if** x **>** 21**:**

**print(**"PERDU"**)**

s**[**1**]** **+=** 0.5

**return** **-**0.5

**else:**

**if** y **>** x**:**

**print(**"PERDU"**)**

s**[**1**]** **+=** 0.5

**return** **-**0.5

**else:**

**if** x **==** y**:**

**print(**"ÉGALITÉ"**)**

**return** 0

**else:**

**print(**"GAGNÉ"**)**

s**[**0**]** **+=** 0.5

**return** 0.5

* Le programme »interface\_graphique.py » :

**import** globalVars

**from** decklist **import** **\***

**from** tkinter **import** **\***

**from** tkinter**.**filedialog **import** **\***

**def** **affiche\_cartes(**hand**,**y**,**cartes**):**#fonction qui affiche les cartes sur la table

a**=**len**(**hand**)** #y est la hauteur ou l'on affiche les cartes

**for** i **in** range**(**a**):**

carte **=** hand**[**i**]**

cartes**.**append**(** PhotoImage**(**file **=** "..\\cartes\\" **+** carte **+**".gif"**)** **)**

canvas**.**create\_image**((**10**+**100**\***i**),** y**,** image **=** cartes**[**i**],** anchor **=** NW**)**

**def** **affiche\_table():**#fonction qui affiche les objets de la table

**del** cartes\_joueur**[:]**

**del** cartes\_croupier**[:]**

canvas**.**delete**(**ALL**)**#enleve les cartes de la table pour en remettre après

affiche\_cartes**(**hand\_joueur**,** 300**,** cartes\_joueur**)**

affiche\_cartes**(**hand\_croupier**,** 40**,** cartes\_croupier**)**

canvas**.**create\_text**(**50**,** 500**,** text **=** "Vous avez "**+**str**(**total**(**hand\_joueur**)),** fill **=** "white"**)**# Le total de la main du joueur

canvas**.**create\_text**(**50**,** 20**,** text **=** "Le croupier a "**+**str**(**total**(**hand\_croupier**)),** fill **=** "white"**)**#Le total de la main du croupier

#afficher le compte et les scores

textCompte**.**set**(**"Compte : "**+**str**(**globalVars**.**compte**))**

textScore\_joueur**.**set**(**"Votre score : "**+**str**(**globalVars**.**Scores**[**0**]))**

textScore\_croupier**.**set**(**"Score du croupier : "**+**str**(**globalVars**.**Scores**[**1**]))**

fenetre**.**update\_idletasks**()**

**def** **rejouer():**#fonction du bouton rejouer

#remet les mains et le total des mains à 0

**del** hand\_croupier**[:]**

**del** hand\_joueur**[:]**

#distribue des nouvelles cartes et les affiche

tirer**(**hand\_croupier**,**1**)**

tirer**(**hand\_joueur**,** 2**)**

affiche\_table**()**

bouton\_rejouer**.**config**(**state**=**DISABLED**)**

bouton\_carte**.**config**(**state**=**ACTIVE**)**

bouton\_rester**.**config**(**state**=**ACTIVE**)**

**if** total**(**hand\_joueur**)** **>=** 21**:**

bouton\_carte**.**config**(**state**=**DISABLED**)**

rester**()**

**def** **carte():**#fonction du bouton carte

tirer**(**hand\_joueur**,**1**)**

**if** total**(**hand\_joueur**)** **>=** 21**:**

bouton\_carte**.**config**(**state**=**DISABLED**)**

rester**()**

affiche\_table**()**

**def** **rester():**#fonction du bouton rester

croupier**(**hand\_croupier**)**

gagnant**(**total**(**hand\_joueur**),**total**(**hand\_croupier**),**globalVars**.**Scores**)**

affiche\_table**()**

bouton\_rejouer**.**config**(**state**=**ACTIVE**)**

bouton\_carte**.**config**(**state**=**DISABLED**)**

bouton\_rester**.**config**(**state**=**DISABLED**)**

winner **=** gagnant**(**total**(**hand\_joueur**),**total**(**hand\_croupier**),**globalVars**.**Scores**)**

**print(**winner**)**

**if** winner **==** 0**:**

canvas**.**create\_text**(**260**,**260**,** text **=** "ÉGALITÉ" **,** fill**=**"white"**)**

**elif** winner **==** 0.5**:**

canvas**.**create\_text**(**260**,**260**,** text **=** "GAGNÉ" **,** fill**=**"white"**)**

**else:**

canvas**.**create\_text**(**260**,**260**,** text **=** "PERDU" **,** fill**=**"white"**)**

fenetre**.**update\_idletasks**()**

hand\_croupier **=** **[]**

hand\_joueur **=** **[]**

cartes\_joueur **=** **[]**

cartes\_croupier **=** **[]**

fenetre **=** Tk**()** #fenetre principale

textCompte **=** StringVar**()**

textScore\_joueur **=** StringVar**()**

textScore\_croupier **=** StringVar**()**

champ\_label **=** Label**(**fenetre**,** text**=**"Jeu de Blackjack"**)**

champ\_label**.**pack**()**

table **=** Frame**(**fenetre**,** width**=**1**,** height**=**1**,** borderwidth**=**0**,** bg**=**"#1D702D"**)**#cadre ou il y aura les cartes

table**.**pack**(**side **=** RIGHT**)**

bouton\_carte **=** Button**(**fenetre**,** text**=**"Carte"**,** command**=**carte**)**#bouton pour piocher une carte

bouton\_rester **=** Button**(**fenetre**,** text**=**"Rester"**,** command**=**rester**)**#bouton pour s'arrêter

bouton\_carte**.**config**(**state**=**DISABLED**)**

bouton\_rester**.**config**(**state**=**DISABLED**)**

bouton\_carte**.**pack**(**pady**=**10**)**

bouton\_rester**.**pack**(**pady**=**10**)**

message1 **=** Label**(**fenetre**,** textvariable **=** textScore\_joueur**)**#Scores

message1**.**pack**(**pady**=**10**)**

message2 **=** Label**(**fenetre**,** textvariable **=** textScore\_croupier**)**

message2**.**pack**(**pady**=**10**)**

message3 **=** Label**(**fenetre**,** textvariable **=** textCompte**)**#compte des cartes

message3**.**pack**(**pady**=**10**)**

canvas **=** Canvas**(**table **,**width **=** 500**,** height **=** 520**,** bg **=** "#1D702D"**)**#emplacement des cartes

canvas**.**pack**()**

bouton\_rejouer **=** Button**(**fenetre**,** text**=**"Rejouer" **,** command **=** rejouer**)**#bouton rejouer

bouton\_rejouer**.**pack**(**side**=**"top"**)**

bouton\_quitter **=** Button**(**fenetre**,** text**=**"Quitter"**,** command **=** fenetre**.**destroy**)**#bouton quitter

bouton\_quitter**.**pack**(**side**=**"bottom"**)**

fenetre**.**mainloop**()**

A ces 3 fichiers python s’ajoutent des images au format GIF qui sont les images appelées dans l’interface graphique (images des cartes et tableau de stratégie de base).