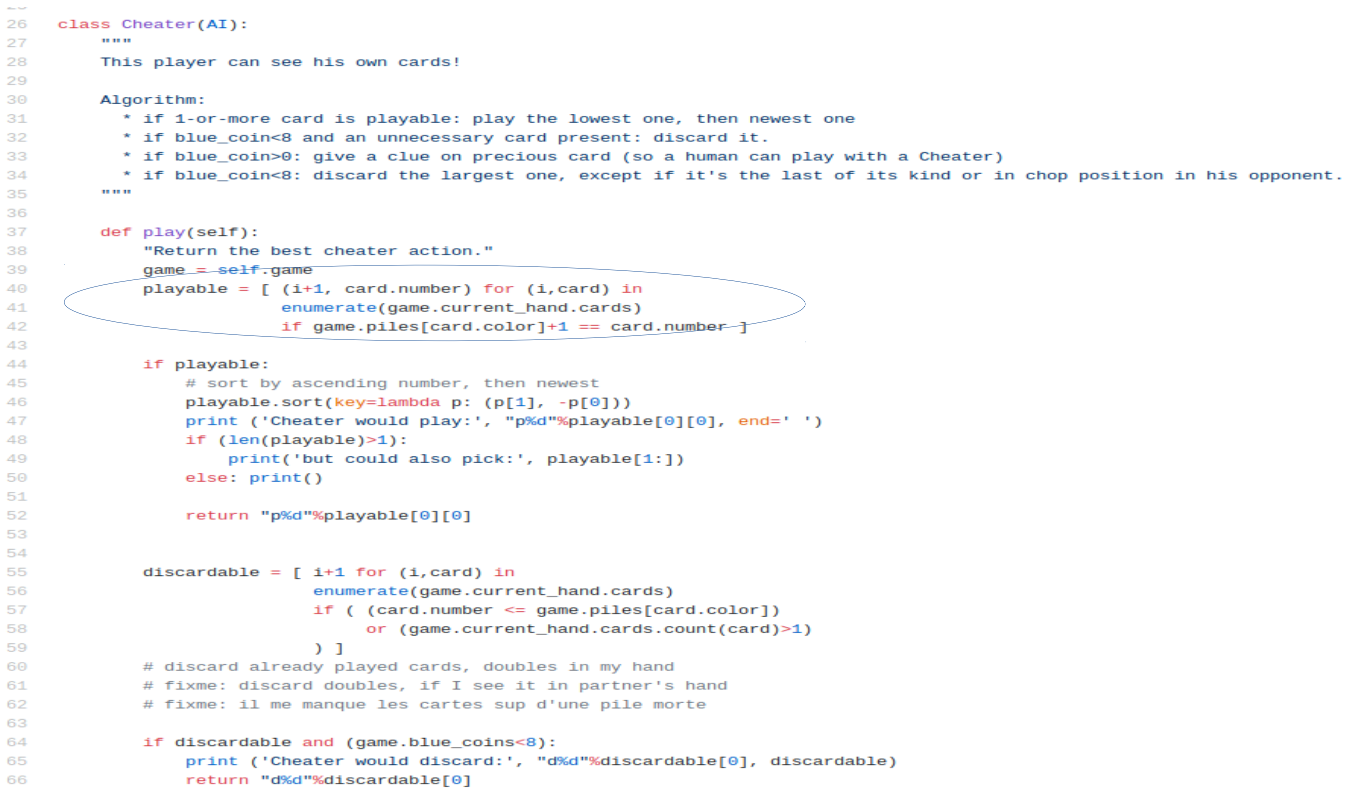
Projet d'IN 104 : Hanabi

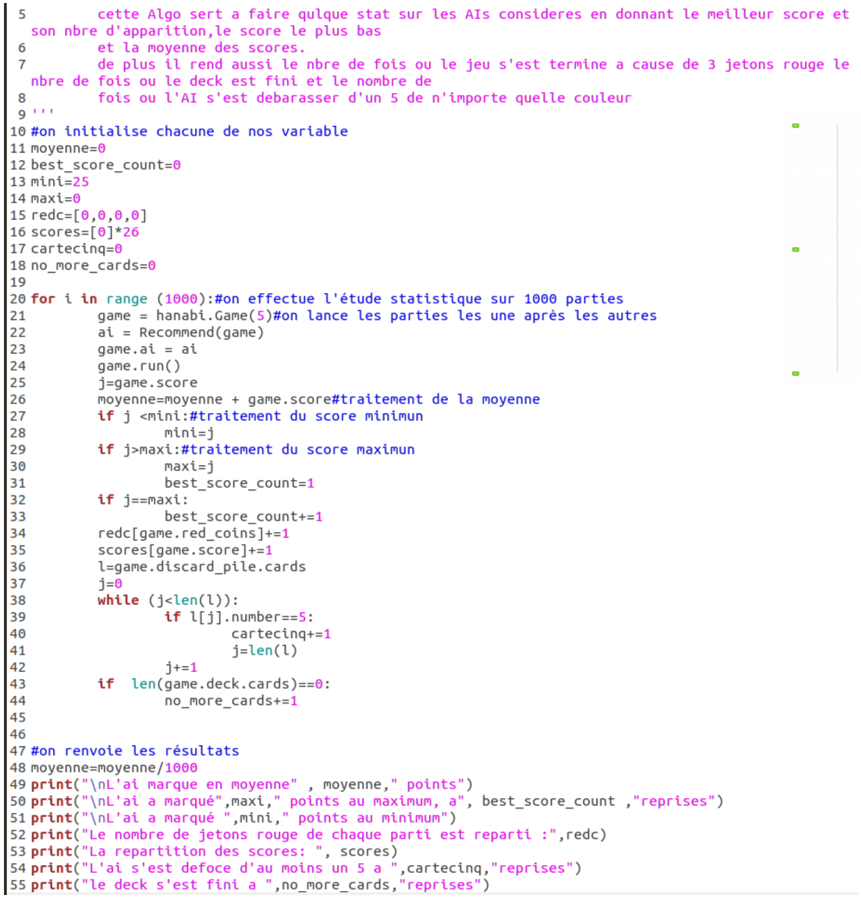
Charbel ALHELOU – Luc LAPORTE

I-Point de départ : l’AI cheater

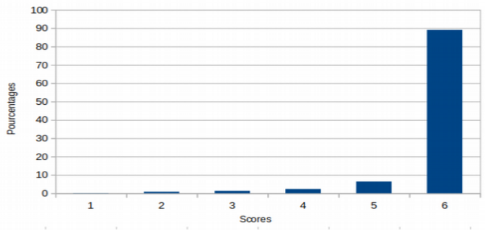


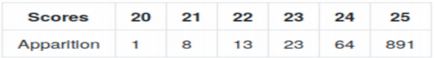
```python
class Cheater(AI):
    """
    This player can see his own cards!

    Algorithm:
      * if 1-or-more card is playable: play the lowest one, then newest one
      * if blue_coin<8 and an unnecessary card present: discard it.
      * if blue_coin>0: give a clue on precious card (so a human can play with a Cheater)
      * if blue_coin<8: discard the largest one, except if it's the last of its kind or in chop position in his opponent.
    """

    def play(self):
        "Return the best cheater action."
        game = self.game
        playable = [ (i+1, card.number) for (i,card) in
                     enumerate(game.current_hand.cards)
                     if game.piles[card.color]+1 == card.number ]

        if playable:
            # sort by ascending number, then newest
            playable.sort(key=lambda p: (p[1], -p[0]))
            print ('Cheater would play:', "p%d"%playable[0][0], end=' ')
            if (len(playable)>1):
                print('but could also pick:', playable[1:])
            else: print()

            return "p%d"%playable[0][0]


        discardable = [ i+1 for (i,card) in
                        enumerate(game.current_hand.cards)
                        if ( (card.number <= game.piles[card.color])
                             or (game.current_hand.cards.count(card)>1)
                        ) ]
        # discard already played cards, doubles in my hand
        # fixme: discard doubles, if I see it in partner's hand
        # fixme: il me manque les cartes sup d'une pile morte

        if discardable and (game.blue_coins<8):
            print ('Cheater would discard:', "d%d"%discardable[0], discardable)
            return "d%d"%discardable[0]
```

I-Point de départ : l’AI cheater



```
        cette Algo sert a faire qulque stat sur les AIs consideres en donnant le meilleur score et son nbre d'apparition,le score le plus bas
        et la moyenne des scores.
        de plus il rend aussi le nbre de fois ou le jeu s'est termine a cause de 3 jetons rouge le nbre de fois ou le deck est fini et le nombre de
        fois ou l'AI s'est debarasser d'un 5 de n'importe quelle couleur
'''
#on initialise chacune de nos variable
moyenne=0
best_score_count=0
mini=25
maxi=0
redc=[0,0,0,0]
scores=[0]*26
cartecinq=0
no_more_cards=0

for i in range (1000):#on effectue l'étude statistique sur 1000 parties
        game = hanabi.Game(5)#on lance les parties les une après les autres
        ai = Recommend(game)
        game.ai = ai
        game.run()
        j=game.score
        moyenne=moyenne + game.score#traitement de la moyenne
        if j <mini:#traitement du score minimun
                mini=j
        if j>maxi:#traitement du score maximun
                maxi=j
                best_score_count=1
        if j==maxi:
                best_score_count+=1
        redc[game.red_coins]+=1
        scores[game.score]+=1
        l=game.discard_pile.cards
        j=0
        while (j<len(l)):
                if l[j].number==5:
                        cartecinq+=1
                        j=len(l)
                j+=1
        if  len(game.deck.cards)==0:
                no_more_cards+=1


#on renvoie les résultats
moyenne=moyenne/1000
print("\nL'ai marque en moyenne" , moyenne," points")
print("\nL'ai a marqué",maxi," points au maximum, a", best_score_count ,"reprises")
print("\nL'ai a marqué ",mini," points au minimum")
print("Le nombre de jetons rouge de chaque parti est reparti :",redc)
print("La repartition des scores: ", scores)
print("L'ai s'est defoce d'au moins un 5 a ",cartecinq,"reprises")
print("le deck s'est fini a ",no_more_cards,"reprises")
```

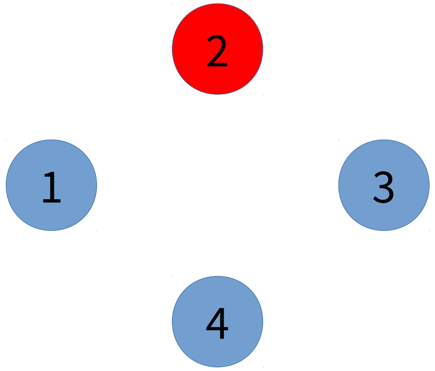


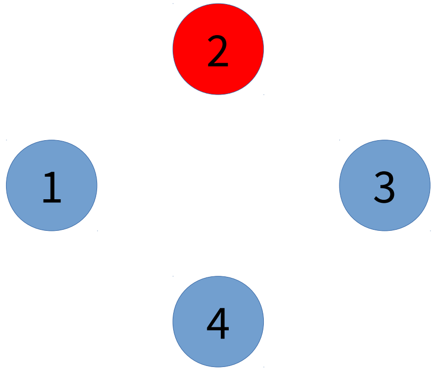


Scores	20	21	22	23	24	25
Apparition	1	8	13	23	64	891

III-Hat guessing AI

Jeu du Hat guessing :





1 voit 2 bleu = Devine Bleu

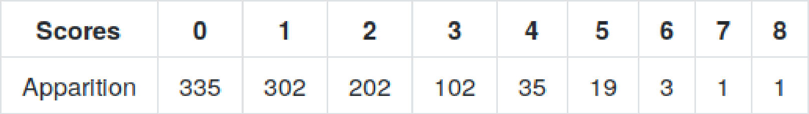
2 sait que 1 voit 2 bleu

2 devine Rouge

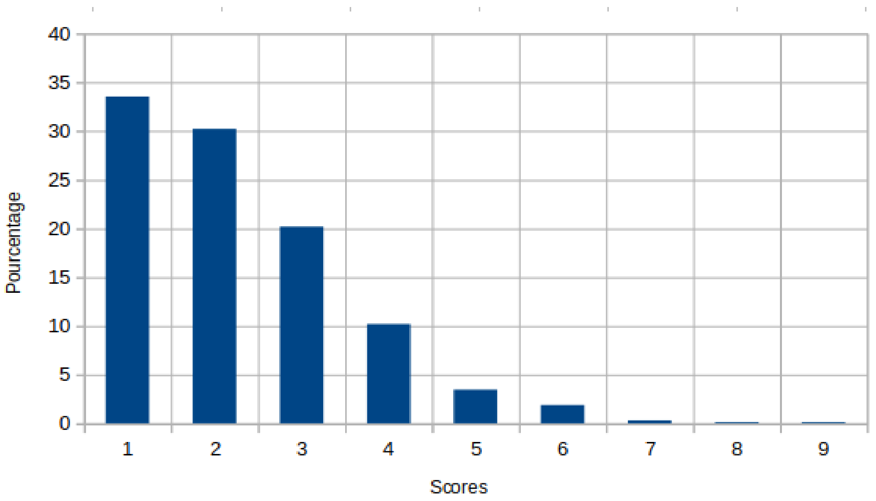
...

Réussite de 3.5 % car 1 se sacrifie

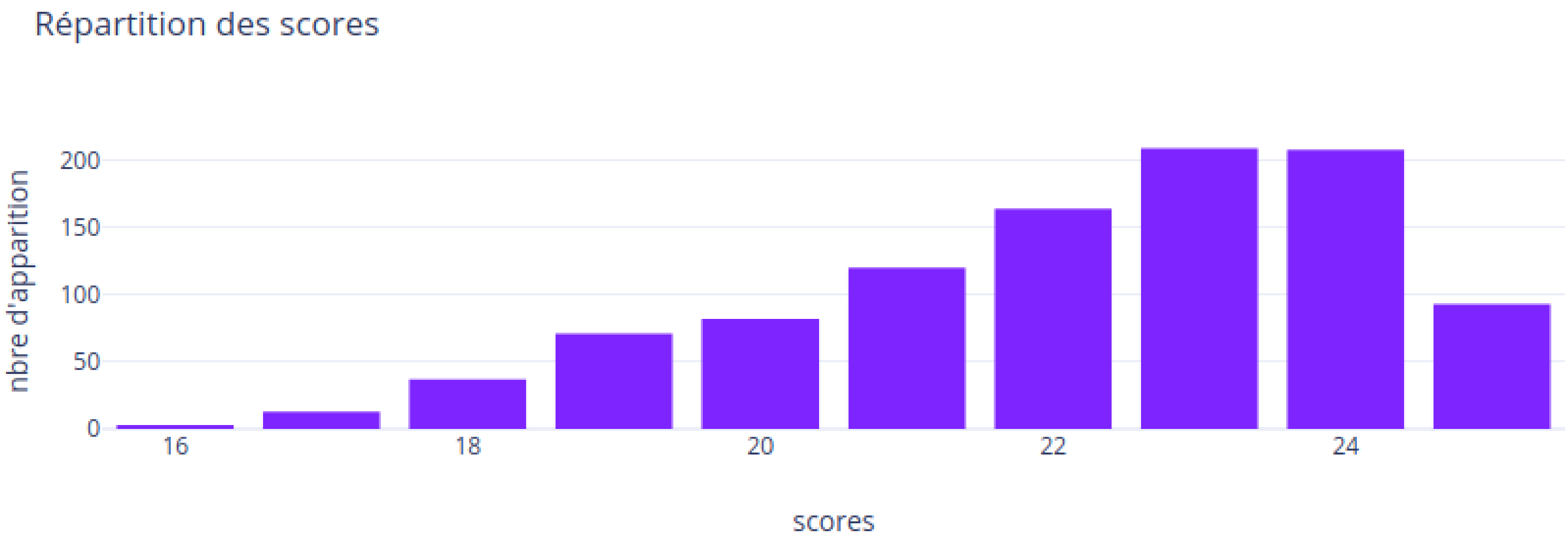
II-Premier Pas : AI aléatoire



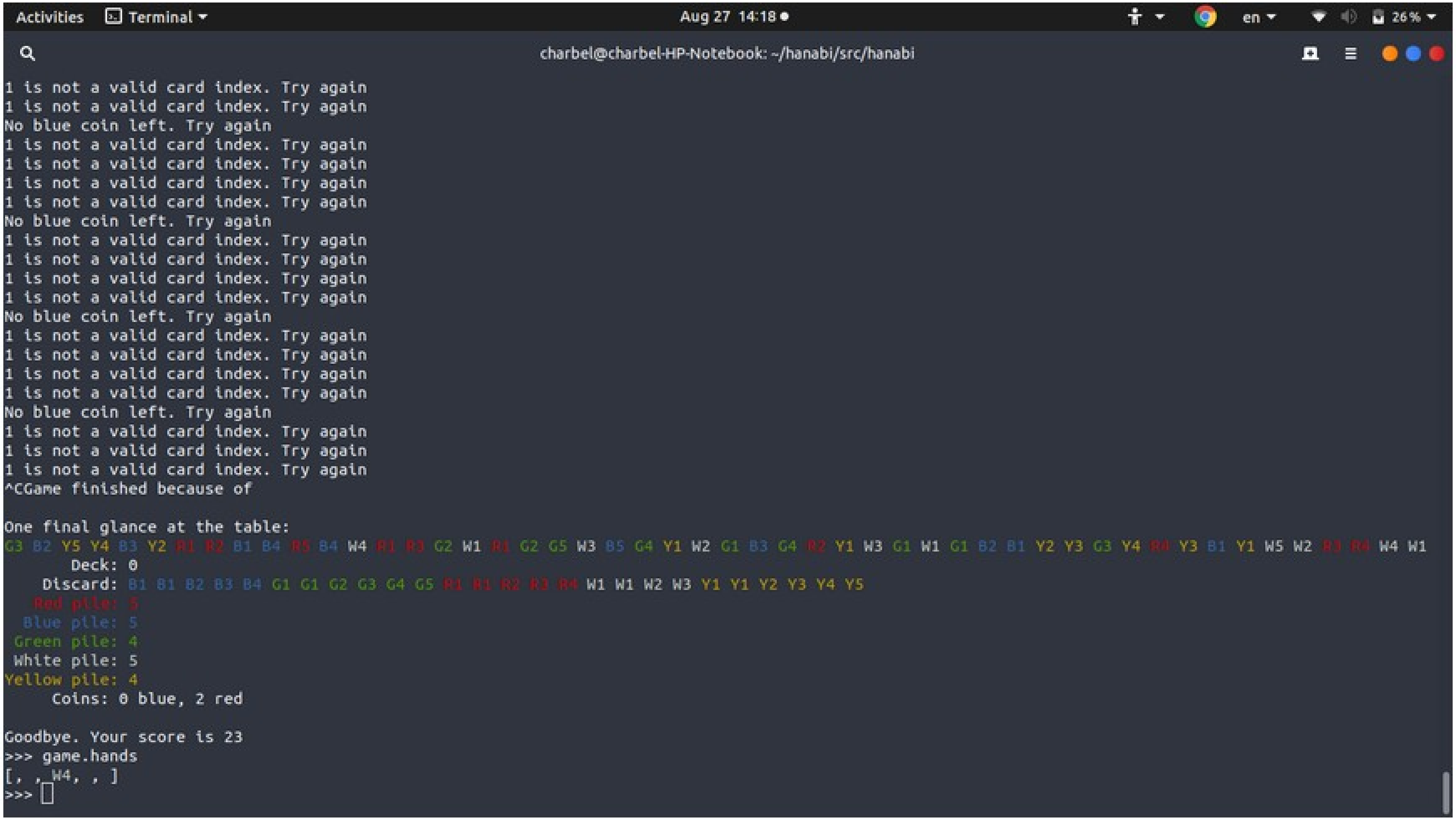
Scores	0	1	2	3	4	5	6	7	8
Apparition	335	302	202	102	35	19	3	1	1



III- Hat guessing AI



III-Hat guessing AI



```
1 is not a valid card index. Try again
1 is not a valid card index. Try again
No blue coin left. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
No blue coin left. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
No blue coin left. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
No blue coin left. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
1 is not a valid card index. Try again
^CGame finished because of

One final glance at the table:
G3 B2 Y5 Y4 B3 Y2 R1 R2 B1 B4 R5 B4 W4 R1 R3 G2 W1 R1 G2 G5 W3 B5 G4 Y1 W2 G1 B3 G4 R2 Y1 W3 G1 W1 G1 B2 B1 Y2 Y3 G3 Y4 R4 Y3 B1 Y1 W5 W2 R3 R4 W4 W1
       Deck: 0
    Discard: B1 B1 B2 B3 B4 G1 G1 G2 G3 G4 G5 R1 R1 R2 R3 R4 W1 W1 W2 W3 Y1 Y1 Y2 Y3 Y4 Y5
   Red pile: 5
  Blue pile: 5
 Green pile: 4
 White pile: 5
Yellow pile: 4
      Coins: 0 blue, 2 red

Goodbye. Your score is 23
>>> game.hands
[, , W4, , ]
>>> 
```