BOUHEDDA Bilel
NGUYEN Simon

S1-C1

SAÉ R102
The Missing Token Game
Artificial Intelligence (AI)

IUT
Belfort-
Montbéliard

INFO
Département Informatique
IUT Belfort-Montbéliard

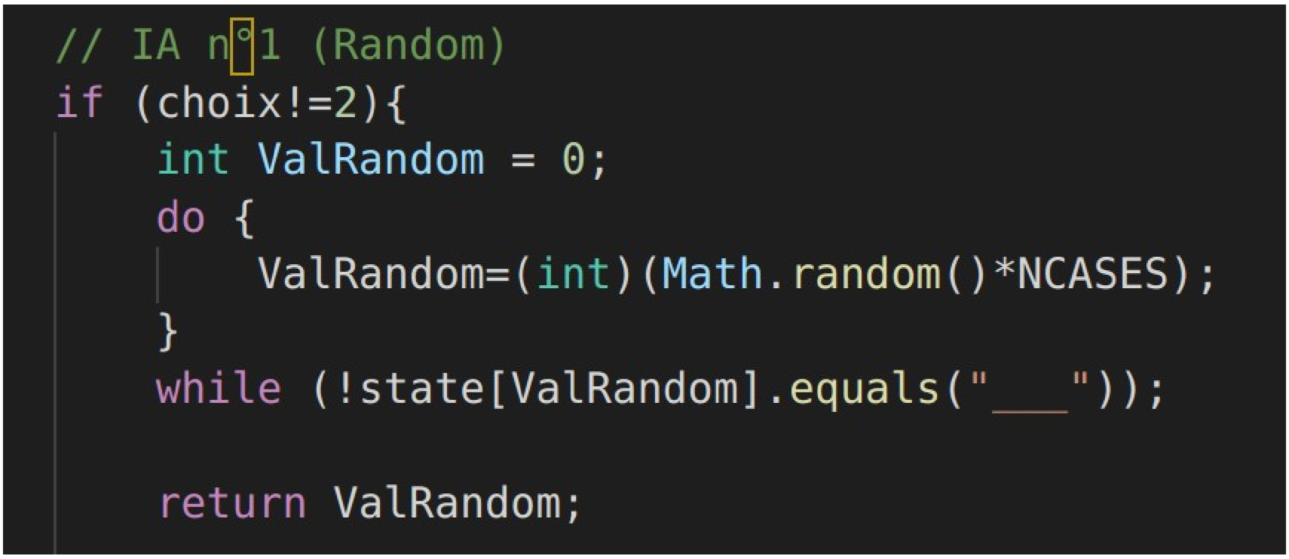
Introduction

First of all, Artificial intelligence (AI for Artificial Intelligence, or IA in French) consists in implementing a certain number of techniques aimed at allowing machines to imitate a form of real intelligence. Firstly, we coded an AI called Random (naive) to test our first tests with an AI then we thought of an Artificial Intelligence capable of having a better strategy than the AI Random by making several games between us. After having introduced you to our two Artificial Intelligences, we will compare and analyze them.

1. Presentation of Artificial Intelligence

1.1. Random Artificial Intelligence

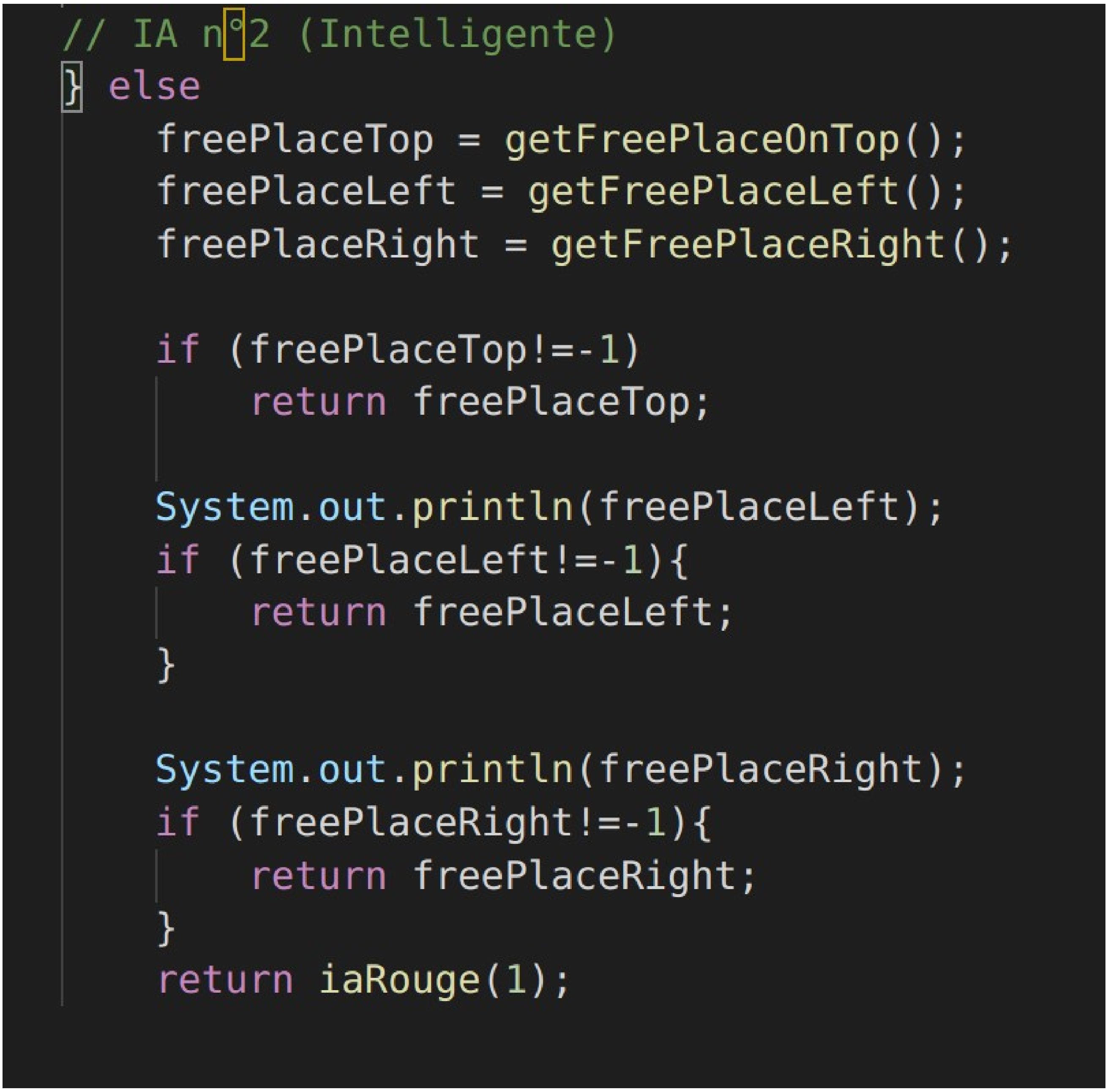
As said in the name, random artificial intelligence returns a random value. This value is tested in the condition of a while loop so as not to find a value already taken before. That is, a different value used in the state array (an empty location).



```
// IA n°1 (Random)
if (choix!=2){
    int ValRandom = 0;
    do {
        ValRandom=(int)(Math.random()*NCASES);
    }
    while (!state[ValRandom].equals("___"));

    return ValRandom;
```

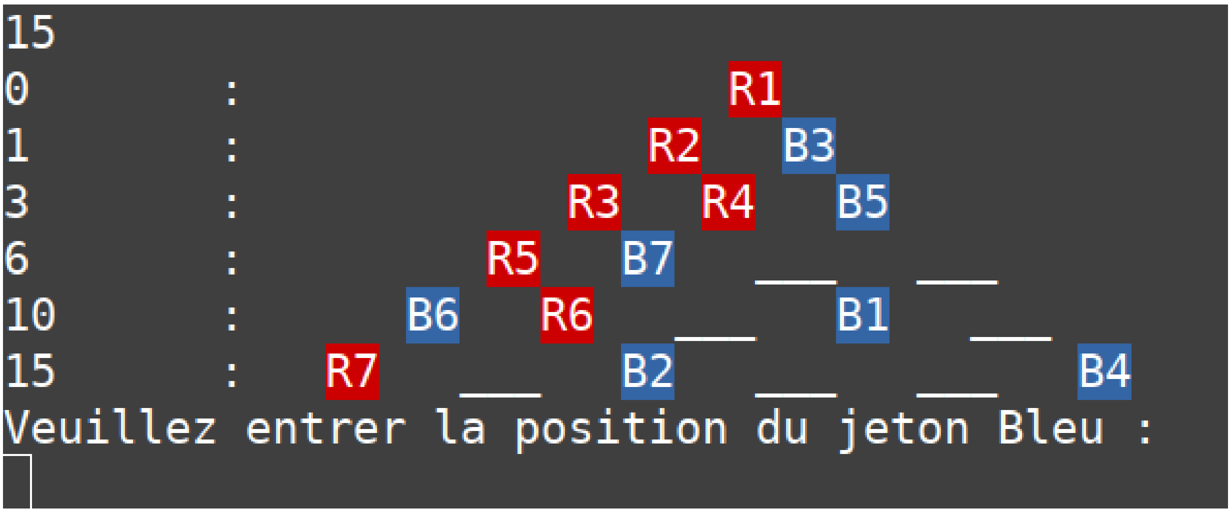
1.2. Artificial intelligence not naive

The principle of the second Artificial Intelligence is simple, it focuses only on the corners of the game (top, left and right), we thought of an optimized strategy, after several test games between us, we found that if we played in the center, we had a better chance of losing the game. so we made an ai that mainly plays in the corners of the game.



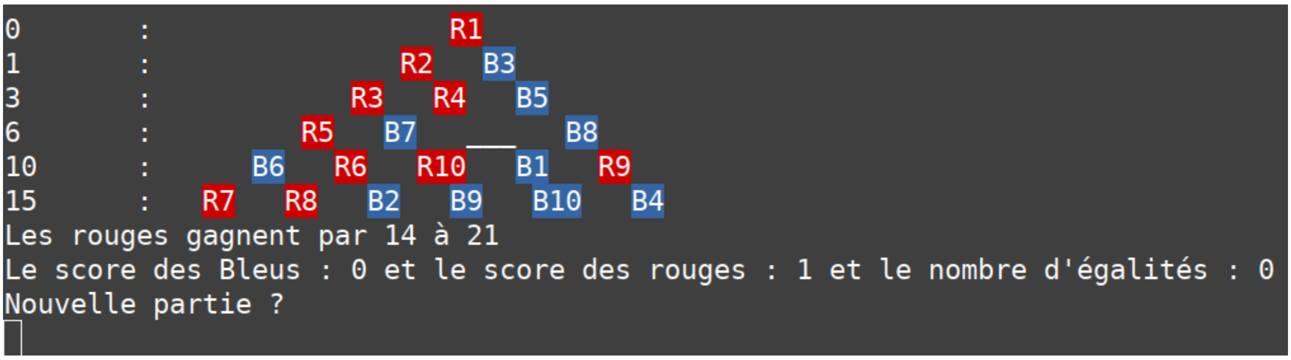
```
// IA n°2 (Intelligente)
} else
    freePlaceTop = getFreePlaceOnTop();
    freePlaceLeft = getFreePlaceLeft();
    freePlaceRight = getFreePlaceRight();

    if (freePlaceTop!=-1)
        return freePlaceTop;

    System.out.println(freePlaceLeft);
    if (freePlaceLeft!=-1){
        return freePlaceLeft;
    }

    System.out.println(freePlaceRight);
    if (freePlaceRight!=-1){
        return freePlaceRight;
    }
    return iaRouge(1);
```

Artificial Intelligence is composed of three external functions, the first function "getFreePlaceOnTop ()", consists in checking if there are places available only in the corner at the top of the game. If the function returns an error, that is say that there is no more space in the top corner, so the IA will use the second function "getFreePlaceLeft ()", with the same principle as the first function "getFreePlaceOnTop ()" but in the corner at left of the game. If this function returns an error, it means that there is no more space left in the left corner either. Then the IA uses the "getFreePlaceRight ()" function with the same principle but in the right corner of the game. If the game still continues after all three corners are no longer available, the IA does random values.



```
15
0       :                  R1
1       :               R2   B3
3       :            R3   R4   B5
6       :         R5   B7   ___   ___
10      :      B6   R6   ___   B1   ___
15      :   R7   ___   B2   ___   ___   B4
Veuillez entrer la position du jeton Bleu :
```

Here is an example of a game with a human against artificial intelligence (not naive), so we can see that the AI is playing at first, in the top right corner, the blue token "B3" is in the way. AI however, AI continues to play the next place. When the "B5" token is placed, the first "getFreePlaceOnTop ()" function returns an error, the AI will use the second "getFreePlaceLeft ()" function to complete the left corner of the game.



```
0       :                  R1
1       :               R2   B3
3       :            R3   R4   B5
6       :         R5   B7   ___   B8
10      :      B6   R6   R10   B1   R9
15      :   R7   R8   B2   B9   B10   B4
Les rouges gagnent par 14 à 21
Le score des Bleus : 0 et le score des rouges : 1 et le nombre d'égalités : 0
Nouvelle partie ?
```

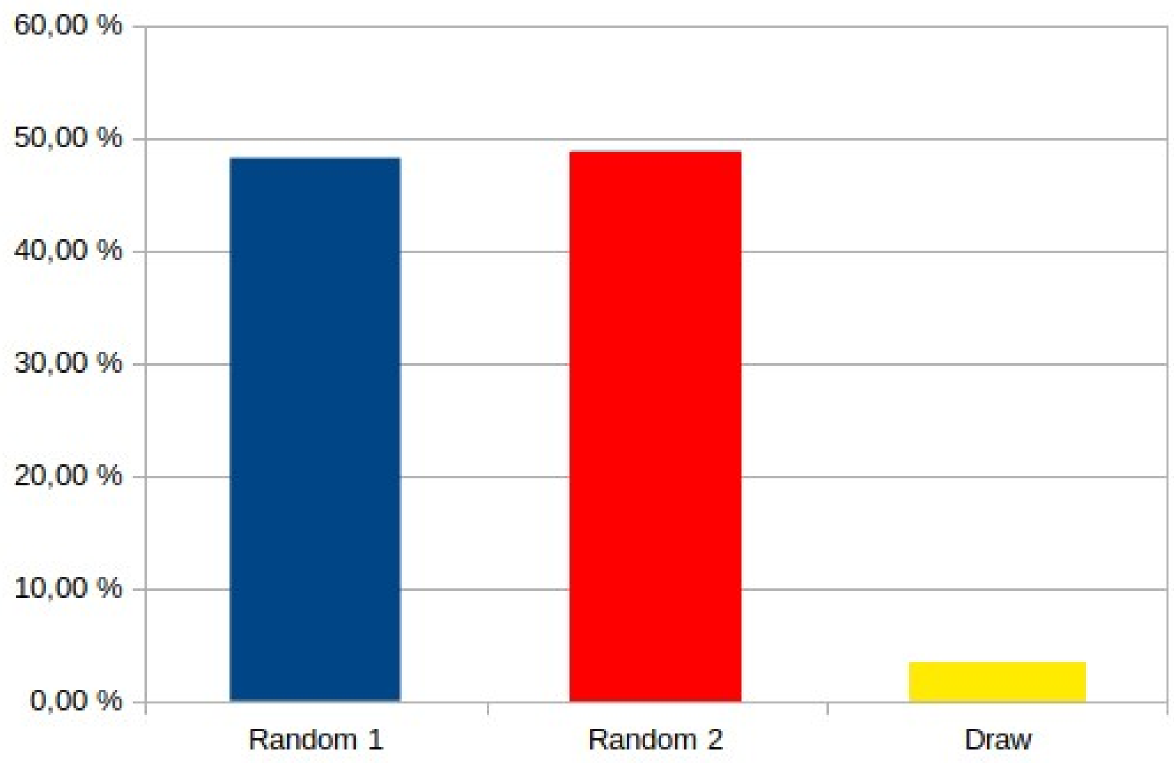
In this situation, there is no more room in the corners of the game, so the Artificial Intelligence will make a random value for the next tokens like the "R10" token.

2. Comparaison between the Artificial Intelligence

Now that we have presented our two artificial intelligences, we have decided to compare them with each other, in order to prove that they are working well.

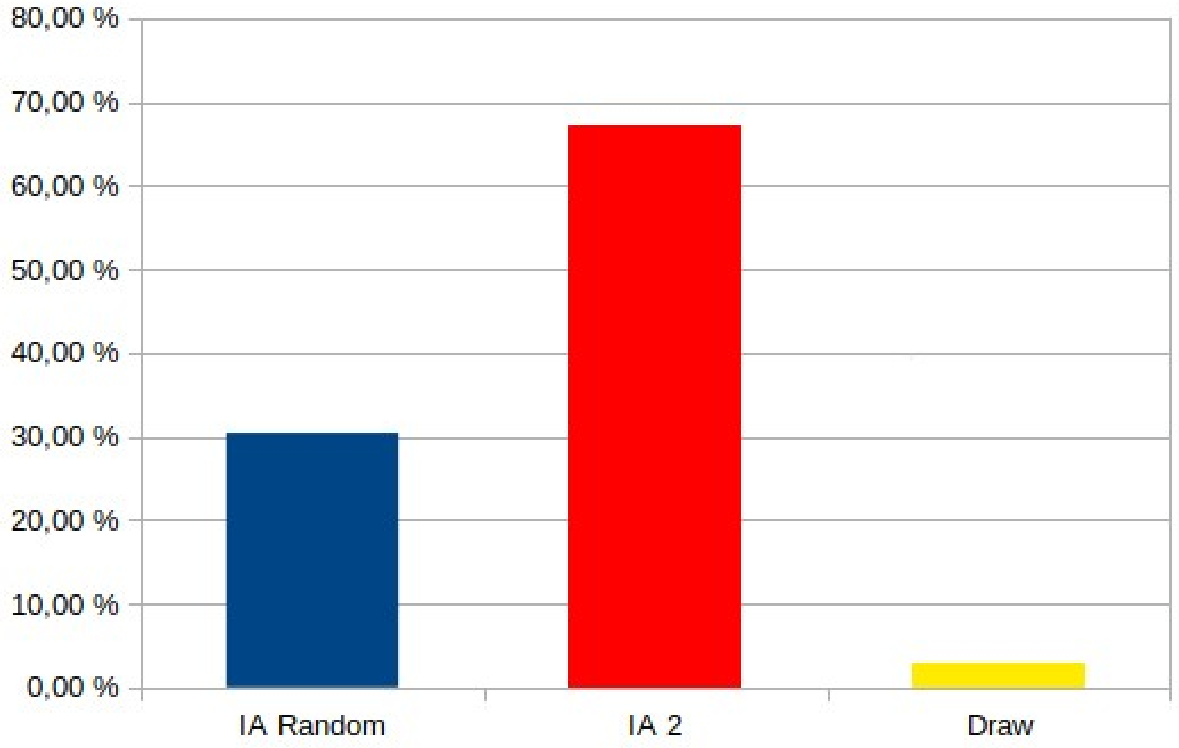
2.1. Random AI vs Random AI

To begin with, we decided to pit two intelligences that work randomly against each other. As expected, we arrive at very tight results, which proves the effectiveness of so-called random artificial intelligences.



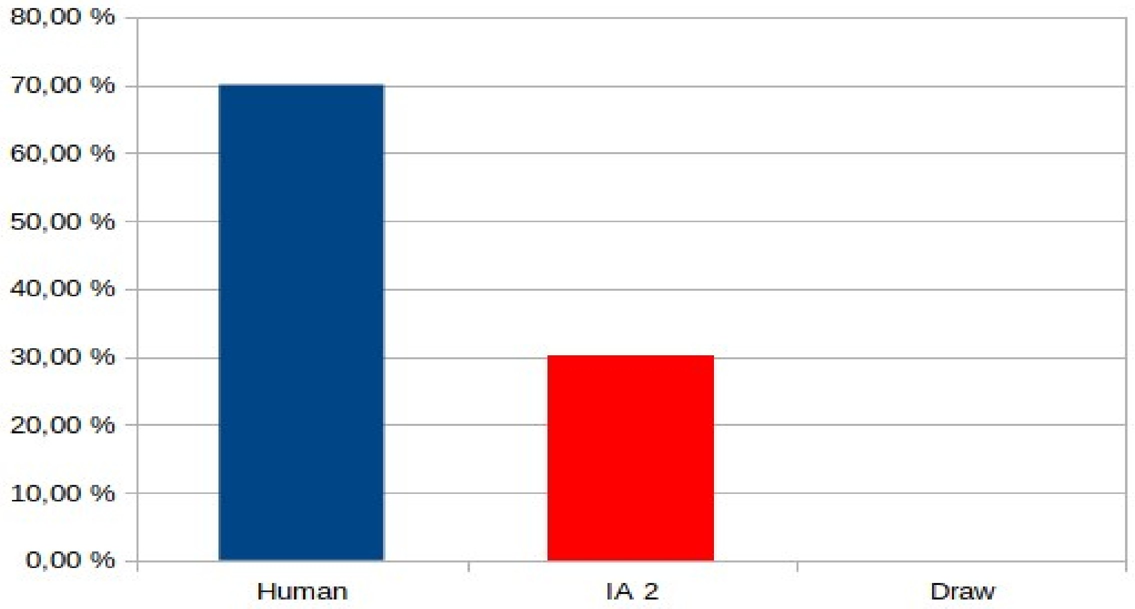
2.2. Random AI vs Not Naive IA

Then, we will face our non-naive AI that we have improved against a random AI. So, we decided to make about 1000 clashes between the two Artificial intelligences, so that the results are very diverse and varied. As can be seen from the graph below, non-naive AI has a win rate of around 70 percent. Against a win rate of around 30 percent for random artificial intelligence. We can therefore conclude that against a random artificial intelligence, our new non-naive artificial intelligence is much more efficient and much stronger.



2.3. Not Naive AI vs Human

Finally, we made our non-naive artificial intelligence face off against a human person, to test it in real conditions, with a human brain. Of course, humans have a much higher win rate. However, we could see that in the first clashes, it was artificial intelligence that won over humans. Thus, we can therefore deduce that humans have adapted to the problem, and have succeeded in identifying artificial intelligence.


Human
IA 2
Draw