# DOCUMENTATION DE VALISOACHESS

## PRÉSENTATION

Nom : ValisoaChess
Type : Moteur d'échecs
Développeur : Valisoa Jean Hervé
Début de programmation : 01/04/2017
Langage de programmation : Pascal
Compilateur : FreePascal 3.0.2
Licence : BSD
Version de la première publication : 0.0.2 (30/05/2017)
Version actuelle : 0.0.7 (02/08/2017); gère la nullité par répétition de coup (3fois).
Nombre de critères : 8
Critère rajouté : La nullité par répétition de coup (3 fois).

## I- GENESE DE VALISOACHESS

ValisoaChess est un programme du jeu d'échecs que j'ai créé à partir de 01/04/2017.
En tant que joueur d'échecs (1654 ELO FIDE en Avril 2017) et amateur en programmation, j'ai eu l'idée de créer mon propre jeu d'échecs. Avant de me lancer, j'ai fait des recherches sur internet et j'ai trouvé beaucoup d'exemples mais soient ils sont trop compliqués pour être compris soient leurs codes sont trop longs (> 2 000 lignes). Alors que j'ai voulu avoir un exemple très simple juste pour débuter; un exemple qui montre juste comment deplacer les pièces et comment generer un coup sans se soucier du gain ou des tactiques gagnantes et cela avec le minimum de codes possibles.
Je me suis décidé alors de créer moi-même ce que je cherchais : VALISOACHESS est alors né.

Mon objectif principal était avant tout de créer le plus SIMPLE moteur d'échecs qui soit,en langage Pascal, en utilisant le MINIMUM DE CODE POSSIBLE. Afin que quiconque voudrait débuter au développement d'un jeu d'échecs en langage Pascal, même un débutant en programmation pourrait utiliser VALISOACHESS comme une référence de départ.
Jusqu'à la limite du possible, j'essayerais de maintenir les idées directrices suivantes :
Simple/Minimum de code/Original/Très intelligent/Documenté.
NB :
1- Original pour moi veut dire pas de plagiats. Je ne veux pas de "Copier/Coller" des codes spécifiques à tel ou tel programme d'échecs.
2- Pas de "Copier/Coller" d'algorithmes spécifiques à un programme d'échecs déjà existant même en Open Source.
3- Toutes fois, je ne me priverai pas des algorithmes que tout le monde utilise.
4- Je pourrais utiliser les autres programmes d'échecs, "libres","OpenSource", comme sources d'inspirations(d'idées, d'imaginations) mais jamais comme une source à "Copier/Coller".
5- Je souhaite que VALISOACHESS soit unique de son genre : Original, intelligent et autonome.
6- Mon rêve est qu'à l'avenir VALISOACHESS soit parmi les meilleurs moteurs d'échecs ou qu'il soit le meilleur de tous pourquoi pas .

ValisoaChess, version 0.0.2 (première version), ne gère pas encore la prise en passant, le roque et l'échecs. Il gère tous les autres mouvements de toutes les pièces y compris les captures.
ValisoaChess joue ses coups de façon aléatoire. Toutes fois, il priorise les captures par rapport aux simples deplacements.
Pour jouer, utilisez la notation algebrique du genre e2e4, e7e5 ou g1f3.
Des suggestions, critiques, aides et conseils seront les bienvenus :) !

## II- VARIABLES ET CONSTANTES GLOBALES

J'essaye au mieux de minimiser les variables et constantes globales afin de reduire les riques de bugs mais dans certains cas je vois l'obligation des les utiliser.

**Les valeurs des pièces :**

```
Const
        vide = 0;
        Pion = 1;
        Cavalier = 3;
        Fou = 4;
        Tour = 5;
        Dame = 9
        Roi = 100;
        Bord = 127;
        Echiquier : array[11..88] of string[4] = ('a1','a2','a3','a4','a5','a6','a7','a8',
                                                  'vide','vide','b1','b2','b3','b4','b5','b6','b7','b8',
                                                  'vide','vide','c1','c2','c3','c4','c5','c6','c7','c8',
                                                  'vide','vide','d1','d2','d3','d4','d5','d6','d7','d8',
                                                  'vide','vide','e1','e2','e3','e4','e5','e6','e7','e8',
                                                  'vide','vide','f1','f2','f3','f4','f5','f6','f7','f8',
                                                  'vide','vide','g1','g2','g3','g4','g5','g6','g7','g8',
                                                  'vide','vide','h1','h2','h3','h4','h5','h6','h7','h8');
```

**Les vecteurs de deplacement des pièces :**

vPion : array[1..4] of shortint =(01,02,11,-9); {Vecteur de deplacement d'un pion }
vCavalier : array[1..8] of shortint =(12,21,19,8,-8,-19,-21,-12); {Vecteur de deplacement d'un Cavalier }
vFou : array[1..4] of shortint =(11,-9,9,-11); {etc.}
vTour : array[1..4] of shortint =(1,-1,10,-10);
vRoiDame : Array[1..8] of Shortint =(-1,1,-10,10,11,-11,9,-9);{ Le vecteur de deplacement est le même pour le Roi et la Dame}

```
var
 Trait : shortint; {1 pour les blancs et -1 pour les noirs}
 Piece : array [0..99] of shortint;{[11..88]+vFou[4]=11-11=0;[11..88]+vFou[1]=88+11=99//Garde-fou pour la boucle repeat}
 Machine,Fin,EnPassant,Echecs, PetitRoqueBlanc, GrandRoqueBlanc, PetitRoqueNoir,
 GrandRoqueNoir : boolean;
 Coupep : array[1..2] of string[5]; { Coups de prise en passant}
 RoiBlancbouge, Toura1bouge, Tourh1bouge, RoiNoirbouge, Tourh8bouge, Toura8bouge : boolean;
 CoupStandard : array[1..100] of string[4]; { Sauvegarde tous coups possibles de ValisoaChess }
CaseDepart1 à CaseDepart4, CaseArrivee1 à CaseArrivee4 : byte : { Utilisées dans la procédure Perpetuelle}
```

## III- LES PROCEDURES ET FONCTIONS

1- Procedure Initialisation : Initialisation des pièces, de l'échiquier, de trait...
2- Function GestiondEchecs(Depart,Arrivee : byte) : boolean : Gère l'échecs.
3- Procedure GestiondeRoque(Depart,Arrivee : byte) : Gère le roque.
4- Procedure ActualiseEchiquier(Coup : string) : Actualise,à chaque coup joué, l'état de l'échiquier, le trait, la prise en passant...
5- Procedure Humain : Permet au joueur humain d'entrer son coup.
6- Procedure ValisoaChessCoup : Générateur de Coup de ValisoaChess.
7- Procedure Arbitre : Permet de cadencer le jeu entre ValisoaChess et un joueur humain.
8- Function Evaluation(nbCoup : byte); : Regroupe les criètres d'évaluation d'un coup et renvoi le meilleur coup parmis les coups possibles.
9- Procedure GestiondePromotion(Depart,Arrivee:byte); : gère la promotion de pions en Dame, Tour, Fou ou Cavalier.
10- Procedure Perpetuelle; : Gestion de nullité par répétition de coup 3 fois successif

## IV- LE FONCTIONNEMENT

### LES CRITERES DU CHOIX DE COUP :

1- L'échecs
2- La valeur des pièces à capturer : Lors de capture, il choisit la pièce ayant la plus grande valeur.
3- La capture aléatoire des pièces de même valeur : si les pièces à capturer ont la même valeur.
4- La prise en passant
5- Le roque
6- La promotion de pion en Dame, Tour, Fou et Cavalier
7- La nullité par répétition de coup : un même coup se répète 3 fois successif
8- Le simple deplacement de pièces