

# Les jeux et l'intelligence artificielle

# Le jeu de dames

# Le jeu de dames

Deux variantes :

- Dames américaines/britanniques ou “checkers”
  - Le damier comprend  $8*8$  cases
  - Le nombre de pions au début est de 24 ( $12 * 2$ )
  - La dame ne peut faire de longs déplacements
- Dames françaises (et internationales)
  - Le damier comprend  $10*10$  cases
  - Le nombre de pions au début est de 40
  - La dame peut faire de longs déplacements

# Le jeu de dames Arthur Samuel (1)

- 15 années de travaux de recherche (47 → 62)
- En 1962 il bat Robert Nearnly ex champion du Connecticut
- C'est un travail personnel alors qu'il travaille chez IBM
- Il se fait battre par un autre programme écrit par C Jeasen et Tom R. Truscott en 1977

# Minimax (1)

- Le **théorème du minimax** de John von Neumann a été démontré en 1928
- Il concerne des jeux à deux joueurs
  - à somme nulle : les gains d'un joueur sont exactement l'opposé des gains de l'autre joueur
  - à information complète : chaque joueur connaît précisément ses possibilités d'action et les possibilités d'action de son adversaire

# Minimax (2)

- Le déroulement d'une partie peut être représenté par un **arbre de jeu** :
  - la racine est la situation initiale
  - un nœud est une situation légale du jeu
  - les fils d'un nœud sont les situations que l'on peut atteindre à partir de ce nœud en respectant les règles du jeu
  - une branche est une séquence de coups (légaux) dans la partie

# Minimax (3)

- Les joueurs sont appelés *MAX* et *MIN*
- Par convention *MAX* commence la partie
  - les nœuds à profondeur paire correspondent aux nœuds où *MAX* doit jouer,
  - les nœuds à profondeur impaire correspondent aux nœuds où *MIN* doit jouer
- Les feuilles correspondent aux fins de partie : les états gagnants ou perdants pour *MAX*, ou encore les états bloqués (matches nuls)

# Minimax (4) : jeu « simple »

- Nous pouvons associer une valeur à chaque feuille:
  - une situation gagnante pour *MAX* vaut  $+1$
  - une situation perdante pour *MAX* vaut  $-1$
  - une situation nulle vaut  $0$ .
- Nous associons alors une valeur à chaque nœud présent dans l'arbre :
  - *MAX* choisit la *plus grande* valeur de ses fils
  - *MIN* choisit la *plus petite* valeur de ses fils



# Minimax (5)

Il est rarement possible de développer l'ensemble de l'arbre de jeu

- Calcul du nombre d'états :
  - Morpion : facteur de branchement  $\approx 3$ ,  
nb de demi-coups = 9 au plus  
dimension  $\approx 3^9 = 19683$
  - Echecs : facteur de branchement  $\approx 35$ ,  
nb de demi-coups = 30 au plus  
dimension  $\approx 35^{30}$

# Minimax (6)

Il est donc nécessaire

- De créer l'arbre de jeu à une profondeur fixée
- De disposer d'une fonction d'évaluation pour évaluer les feuilles (qui ne représentent plus une fin de partie), la valeur obtenue sera d'autant plus grande que la situation sera favorable

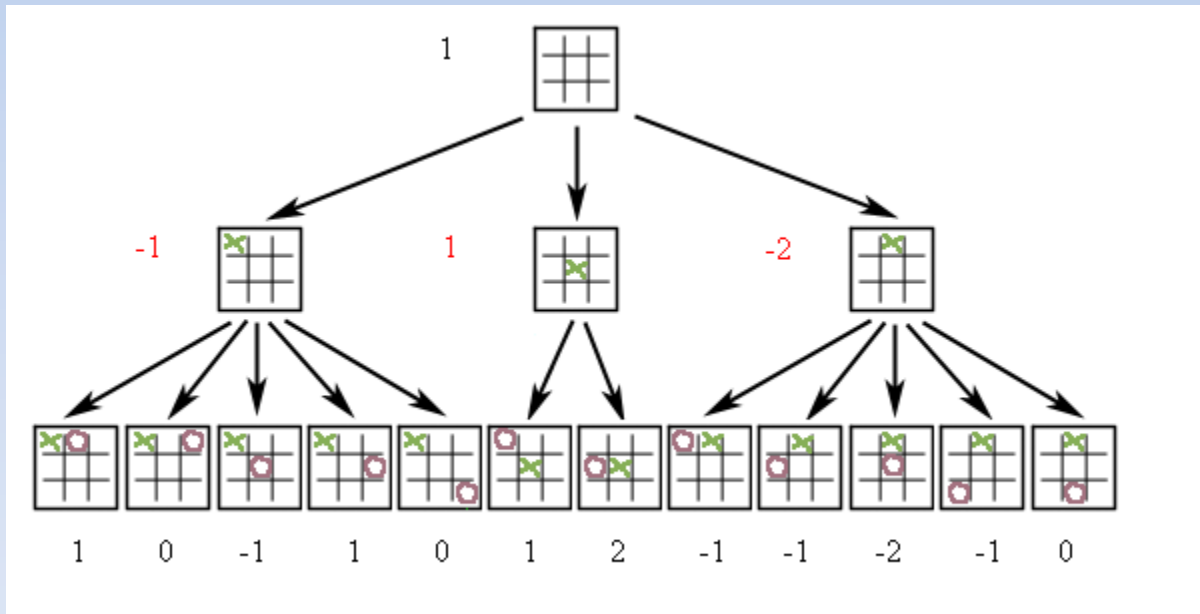
# Minimax (7)

- Exemple de profondeur aux jeu d'échecs (un *coup* signifie en fait un échange, l'action de *MAX* et la réplique de *MIN*)
- 2 coups : niveau novice
- 4 coups : niveau maître
- 6-7 coups : niveau champion du monde.

# Exemple : le morpion

- Nous créons l'arbre de jeu à la profondeur 2
- Nous considérons la fonction d'évaluation suivante :
- $h(\text{situation}) =$ 
  - nb lignes, colonnes, diagonales encore disponibles pour MAX
  - nb lignes, colonnes, diagonales encore disponibles pour MIN

# Le morpion (2)



# Minimax avec coupes $\alpha\beta$ (8)

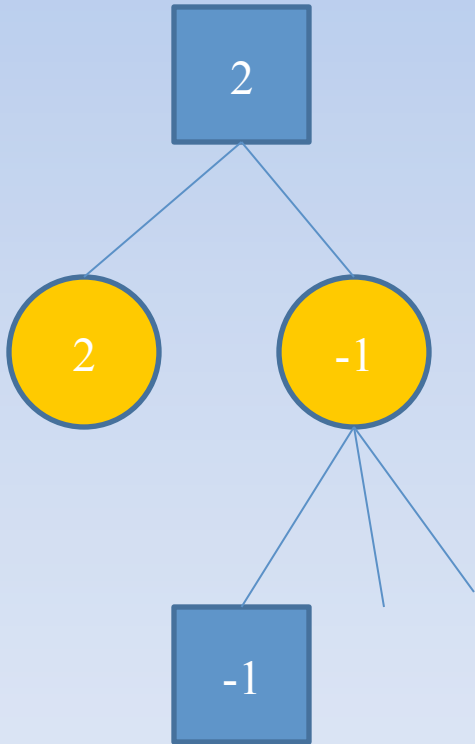
- Sa faiblesse : exploration complète de l'arbre de jeu
- D'où l'idée de ne pas développer entièrement toutes les branches : les branches inutiles sont « coupées » (c'est l'algorithme minimax avec coupes  $\alpha\beta$ ).

# Coupes $\alpha\beta$ (1)

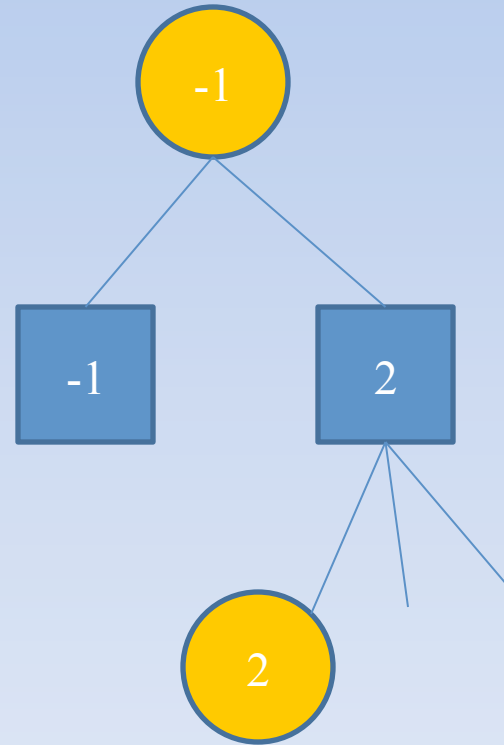
Plusieurs chercheurs inventèrent l'algorithme des coupes  $\alpha\beta$ , on peut citer

- John McCarthy proposa l'idée de Alpha-Beta en **1956** à la conférence de Dartmouth
- Allen Newell et Herbert Simon en donnèrent une approximation en **1958**
- Arthur Samuel en donna une approximation en **1959**
- Daniel Edwards et Timothy Hart le décrivirent en **1961**

# Coupes $\alpha\beta$ (2)



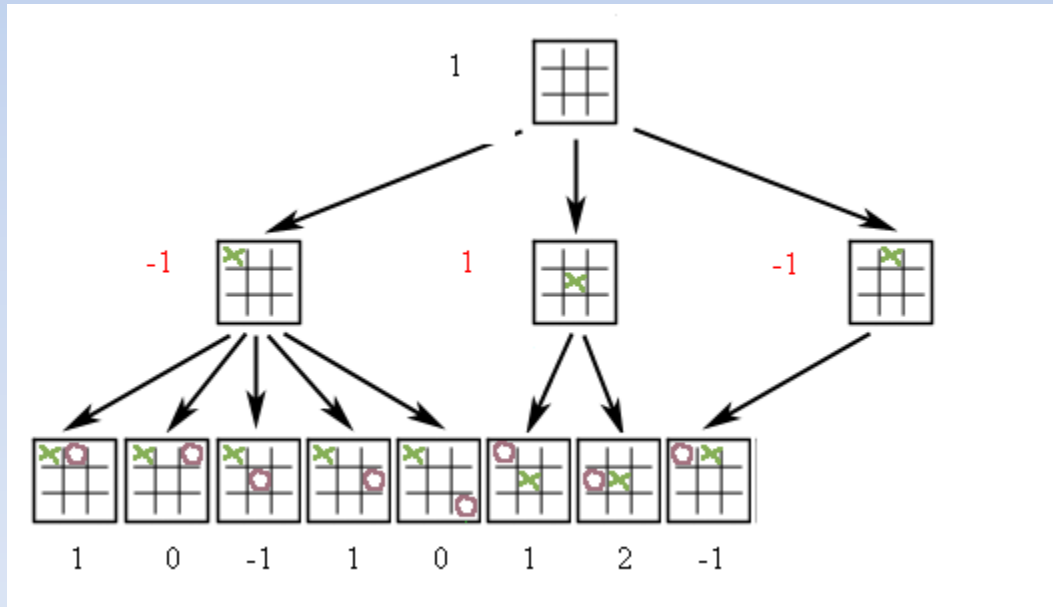
coupe  $\alpha$



coupe  $\beta$



# Le morpion (3)



# La fonction d'évaluation

- Elle dépend de facteurs liés
  - Au matériel (pions,...)
  - A la mobilité (nombre de coups possibles)
  - A la position des pièces
- $F(\text{situation}) =$   
 $P_{\text{matériel}} * f_{\text{matériel}} + P_{\text{mobilité}} * f_{\text{mobilité}} + P_{\text{position}} * f_{\text{position}}$

# L'effet d'horizon

- Le programme ne peut prévoir les effets d'un coup à une profondeur dépassant la profondeur maximum.
- le programme agira pour repousser au-delà de son horizon un événement qui lui est défavorable.
  - le programme repousse les événements défavorables au-delà de son horizon.

# Le jeu de dames – Arthur Samuel

## (2)

- Il utilise l' algorithme minimax avec coupes  $\alpha\beta$   
L'arbre entier contiendrait de l'ordre de  $10^{40}$  nœuds.
- La fonction d'évaluation prend en compte
  - Le nombre de pions et de dames
  - L'occupation des cases centrales
  - L'avance des différents pions
  - Le nombre de coups autorisés
- A Samuel mit au point des techniques statistiques pour attribuer des coefficients à ces différents facteurs

# Chinook (1)



- Début du projet : 1989
- Créé par l'équipe de Jonathan Schaeffer de l'université d'Alberta (Canada)
- Il a perdu en finale du championnat du monde en 1992 contre Marion Tinsley
- Une nouvelle confrontation était prévue en 1994 mais n'a pu être achevée pour des raisons de santé du joueur humain
- Depuis, aucun autre humain n'a pu battre le logiciel

# Chinook (2)



- Schaeffer a mobilisé environ 50 ordinateurs quotidiennement de 1989 à 1990, et parfois jusqu'à 200 ordinateurs

# Chinook (3)

Comment une machine peut-elle battre un joueur ?

- l'ordinateur doit jouer suffisamment bien pour que le joueur humain perde, ou fasse partie nulle, à chaque fois qu'il s'oppose à la machine ( utilisation d'heuristiques).
- Pour gagner infailliblement, un travail d'analyse plus fin est indispensable.

# Chinook (4)

- L'équipe de J. Schaeffer a découvert une stratégie de jeu infailible : l'article est publié en juillet 2007 dans la revue Science, il s'intitule  
« **Checkers is solved** »
- La stratégie obtenue au final fonctionne parfaitement. Que le programme joue en premier (c'est-à-dire Noir) ou qu'il joue en second, jamais il ne perdra, si l'adversaire du programme joue correctement, la partie sera nulle. Deux joueurs parfaits, par exemple l'ordinateur face à lui-même, feront partie nulle.



# Méthode d'analyse rétrograde

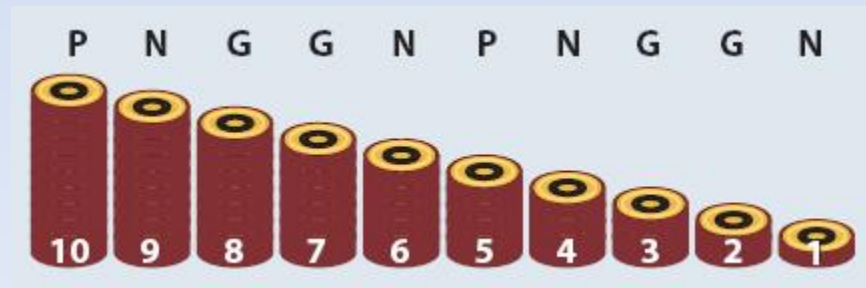
- Les configurations finales correspondant à des parties terminées sont marquées gagnante (pour celui dont c'est le tour de jouer) perdante ou nulle selon les cas.
- Ce marquage permet de connaître la nature perdante, gagnante ou nulle des configurations qui précèdent ce coup final.
- Puis, pas à pas, en remontant, on marque des configurations de plus en plus complexes et proches du début de la partie. Dans le cas des jeux élémentaires, on peut remonter jusqu'à la position initiale dont on sait alors si elle est gagnante, perdante ou nulle.

# Exemple (1)

- Une pile de 10 pions est posée sur une table
  - Deux joueurs jouent à tour de rôle
  - Ils doivent à chaque tour prendre 2 ou 3 pions
  - Ils n'ont pas le droit de passer.
  - Celui qui prend le dernier pion a gagné à moins qu'il ne reste qu'un seul pion, auquel cas la partie est nulle.
- L'analyse rétrograde part des configurations les plus simples et remonte progressivement vers les positions les plus complexes

# Exemple (2)

- La configuration 1 est nulle.
- Les configurations 2 et 3 sont G
- La configuration 4 est nulle
- La configuration 5 est perdante
- La configuration 6 est nulle, etc....



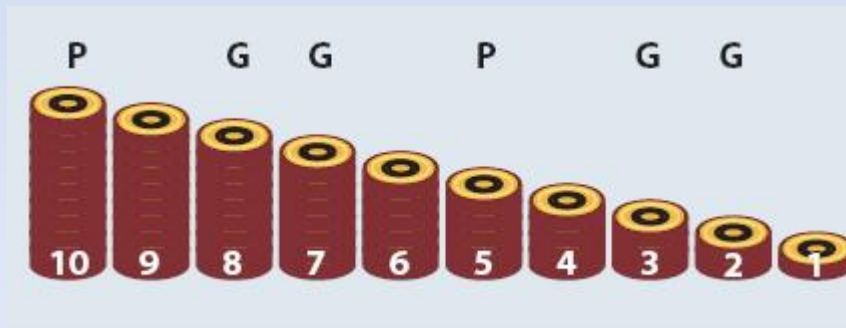
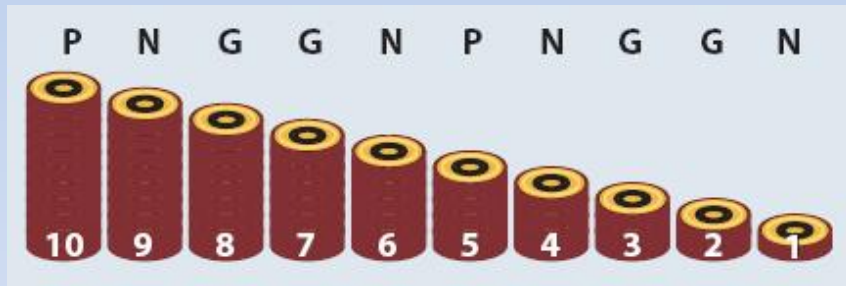
# Les marquages effectués par Chinook

- En 1989, toutes les configurations ayant moins de quatre pièces du jeu étaient marquées et stockées
- En 1996, la base de données des marquages de toutes les configurations à huit pièces ou moins était achevée et utilisée par le programme Chinook
- En 2005 (débuté en 2001), toutes les configurations à 10 pièces sont marquées et enregistrées. La base de données obtenue contient 39 000 milliards de configurations.

# Marquages pour les checkers

- L'équipe de chercheurs de l'Université d'Alberta a réussi le marquage de la position initiale du jeu de Dames anglaises.
- Le marquage de la position initiale a été prouvé à N (partie nulle).
- La partie de l'ensemble des configurations marquées pour arriver à ce résultat est infime ( $10^7$  configurations) par rapport à l'ensemble des configurations du jeu ( $5 \times 10^{20}$  configurations)

# Exemple (3)



on peut savoir avec certitude que la position initiale est P en ne connaissant que les marquages de la figure du bas qui assurent

- que chaque P ne donne accès qu'à des G
- qu'à partir de chaque G, on a accès à au moins un P.

# Après le bas, le haut

- La méthode rétrograde complète est impossible, il faut attaquer le problème par l'autre bout : le début de partie.
- Le but est de découvrir une partie de l'arbre de toutes les configurations ne comportant que des positions dont le marquage a été calculé, le tout définissant une stratégie optimale.
- L'arbre partiel finalement marqué et stocké par l'équipe canadienne comporte  $10^7$  configurations, nombre 10 000 milliards de fois plus petit que le nombre de configurations total.

# Fonctionnement (1)

- Un programme nommé contrôleur tente, à partir de la situation de départ, d'engendrer progressivement l'arbre prouvant que cette situation initiale du jeu est gagnante, perdante ou nulle.
- Le contrôleur détermine à chaque étape quelles sont les positions qui méritent d'être évaluées (c'est-à-dire dont il peut être intéressant de trouver le marquage exact)
- Il lance sur ces configurations soigneusement sélectionnées des programmes nommés résolveurs, qui tentent de découvrir le marquage.



# Fonctionnement (2)

- Un premier résolveur est utilisé dans cette recherche : il est construit sur une ancienne version du programme Chinook.
- Si les résultats fournis ne sont pas « sûrs », ils sont transmis au contrôleur. Puis un second résolveur, fondé sur des travaux de recherche de Ayumu Nagai, de l'Université de Tokyo, est appelé.
- L'ensemble des sous-problèmes à résoudre évolue ainsi petit à petit, géré globalement par le contrôleur qui coordonne les tâches confiées aux résolveurs et récupère auprès d'eux des informations pour la construction de la stratégie optimale.

# Fonctionnement (3)

- Première étape : fournir des marquages (pas forcément certains)
- Une seconde étape est nécessaire : elle produit de nouveaux marquages plus sûrs.
- De proche en proche, le contrôleur dispose de marquages parfaits et donc d'une stratégie optimale valide.
- Le calcul qui a abouti au résultat s'est déroulé pendant plusieurs mois ; il était distribué entre une cinquantaine de machines calculant simultanément les explorations nécessaires. L'arbre trouvé permet un jeu parfait et peut être consulté sur :

**<http://chinook.cs.ualberta.ca/users/chinook/index.html>**

# Conclusion sur les checkers

- Une stratégie optimale n'est connue que pour la position standard de début de partie.
- Dans de nombreux tournois entre joueurs humains, les parties ne commencent pas de la position de départ standard, mais d'une position tirée au hasard en faisant faire trois mouvements licites aux pièces.
- Pour devenir invincible lors de ces compétitions, il faudrait déterminer les stratégies optimales d'environ 200 positions, ce qui, d'après J. Schaeffer, pourrait exiger une autre décennie de travail !

# Les échecs

# Les échecs : les premiers pas

- Claude Shannon publia en 1950 un article consacré aux échecs sur ordinateur
- Alain Turing publia en 1953 un article décrivant les parties jouées par un programme simulé manuellement
- Entre 1955 et 1958, les trois premiers programmes apparaissent : ce furent des « joueurs » médiocres

# Premier programme

- Kotok-McCarthy est le premier programme avec un jeu crédible : il est publié en 1962 au MIT
- En 1966-1967, le premier match entre programmes d'échecs voit le jour. Le programme de l'Institut de physique théorique et expérimentale de Moscou triomphe de Kotok-McCarthy de l'université Stanford. Les coups étaient échangés par télégraphe et le match a duré 9 mois

# Premier programme

Machack de Richard Greenblatt (disciple de M Minsky au MIT) en 1967

- Une section de son programme comprenait des stratégies échiquiennes et des informations sur la manière d'engendrer les coups
- Il utilisait des tables de transposition (conservation dans une table de la valeur de chacune des positions rencontrées)
- Selon les normes de l'USCF, le classement de Machack était entre 1400 et 1500 (proche du niveau moyen des membres de l'USCF)

# Classement Elo

- 1000 : Débutant (enfant)
- 1200 : Débutant
- 1400 : Joueur amateur
- 1600 : Bon joueur
- 1800 : Très bon joueur
- 2000 : Niveau national
- 2200 : candidat maître
- 2300 : Maître FIDE
- 2400 : Maître international (~ 2 565 joueurs)
- 2500 : Grand maître international (~ 900 joueurs)
- 2600 : Les 200 meilleurs joueurs mondiaux
- 2700 : Les 30 meilleurs joueurs mondiaux
- 2800 : Seuls Garry Kasparov, Vladimir Kramnik, Veselin Topalv, Viswanathan Anand et Magnus Carlsen ont dépassé les 2 800 points



# Premiers championnats

- En 1970 l'Association of Computing Machinery organise un tournoi :  
six participants
- En 1971 : huit participants
- En 1974, le premier championnat du monde des ordinateurs a lieu à Stockholm : Kaissa remporte le titre

	<u>Année</u>	<u>Lieu</u>	<u>Vainqueur</u>
1	<u>1974</u>	<u>Stockholm</u>	<u>Kaissa</u>
2	<u>1977</u>	<u>Toronto</u>	<u>Chess 4.6</u>
3	<u>1980</u>	<u>Linz</u>	<u>Belle</u>
4	<u>1983</u>	<u>New York</u>	<u>Cray Blitz</u>
5	<u>1986</u>	<u>Cologne</u>	<u>Cray Blitz</u>
6	<u>1989</u>	<u>Edmonton, Canada</u>	<u>Deep Thought</u>
7	<u>1992</u>	<u>Madrid, Espagne</u>	<u>Chess machine, (Gideon)</u>
8	<u>1995</u>	<u>Hong Kong</u>	<u>Fritz</u>
9	<u>1999</u>	<u>Paderborn, Allemagne</u>	<u>Shredder</u>
10	<u>2002</u>	<u>Maastricht, Pays-Bas</u>	<u>Deep Junior</u>
11	<u>2003</u>	<u>Graz, Autriche</u>	<u>Shredder</u>
12	<u>2004</u>	<u>Université Bar-Ilan, Ramat Gan, Israël</u>	<u>Deep Junior</u>
13	<u>2005</u>	<u>Reykjavík, Islande</u>	<u>Zappa</u>
14	<u>2006</u>	<u>Turin, Italie</u>	<u>Junior</u>
15	<u>2007</u>	<u>Amsterdam, Pays-Bas</u>	<u>Rybka</u>

# Premier défi humain

- L'écossais David Lévy, maître international lança le premier défi en 1968 :  
Il paria qu'aucun ordinateur ne le battrait aux échecs dans les dix ans à venir
- Les enchères montèrent (John MacCarthy et Donald Mitchie relevèrent le gant)

# Le résultat du pari

- En 1978 à Toronto, D Levy remporta la rencontre contre le champion du monde électronique Chess 4.5 par  $3^{1/2}$  à  $1^{1/2}$
- En 1984, il remporta la rencontre contre Cray Blitz, champion électronique du moment

# L'année 1977

- Le premier jeu d'échecs électronique, Chess Challenger, est commercialisé.
- l'International Computer Chess Association (ICCA) est créée.
- 1977, Chess devient le premier ordinateur à remporter un tournoi d'échecs majeur.

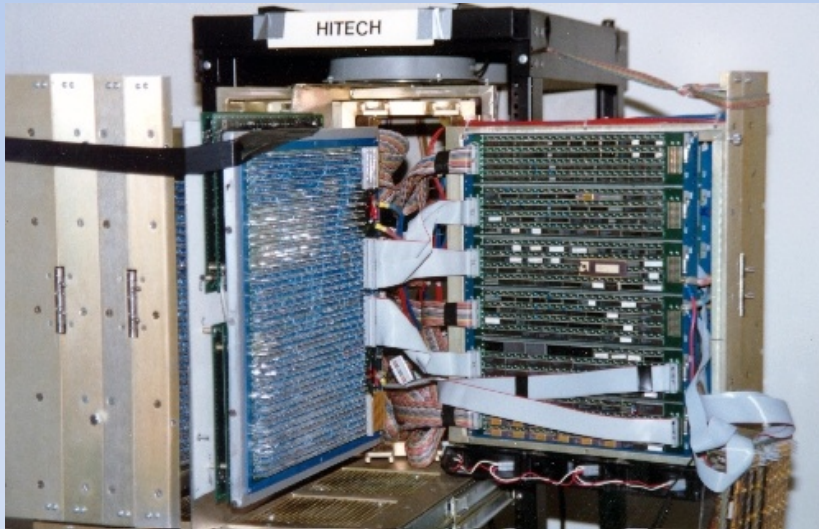
# Le programme Chess

- Créé dans les années 1970 par Larry Atkin et David Slate à l'université Northwestern,
- Il domine les premiers tournois d'échecs des ordinateurs.
- Il est le premier à utiliser la structure bitboard appliquée au jeu d'échecs.
- En 1976, Chess 4.5 remporte la Classe B du tournoi *Paul Masson* de Californie du Nord. Sa performance Elo est alors de 1950
- En 1977, Chess 4.5 remporte l'Open du Minnesota ( 5 victoires et une défaite). Le programme réalise alors une performance Elo de 2271.

# Le programme Belle

- Belle est le nom d'un programme d'échecs et de l'ordinateur (portable) dédié sur lequel il tournait. Il a été développé par Joe Condon et Ken Thompson aux Laboratoires Bell en 1978
- Belle est le premier ordinateur construit dans le seul but de jouer aux échecs
- Il atteint le classement Elo de 2250 et le niveau de maître officiellement en 1983.
- Belle était capable de calculer 160 000 positions par seconde.
- Belle a gagné le championnat du monde d'échecs des ordinateurs en 1980 et le championnat d'échecs Nord-Américain des ordinateurs organisé par l'ACM en 1978, 1980, 1981, 1982 et 1986.

# Le programme Hitech





# Le programme Hitech

- Créé en 1985 par Hans Berliner et Carl Ebeling de Carnegie Mellon University
- Il se composait d'un engin de la taille d'une boîte à chaussures, le « chercheur » relié à une station de travail Sun
- Le chercheur renfermait 64 microprocesseurs consacrés à contrôler chacun une case de l'échiquier
- Hitech descendait en moyenne à la profondeur 8 mais pouvait aller jusqu'à 14

# Le programme Deep Thought

- La version Deep Thought 0.01 a été créée en mai 1988, et la version 0.02 en novembre de la même année
- Les branches les plus prometteuses avaient une profondeur de 15 à 20
- Il analysait 700000 coups par seconde
- Son classement Elo a été estimé à 2551, soit le niveau d'un grand maître international
- Deep Thought a facilement été battu sur les 2 parties d'un match avec Garry Kasparov en 1989

# Deep blue

- Le projet ChipTest est lancé par les étudiants Feng-hsiung Hsu, Murray Campbell et Thomas Anantharaman au laboratoire de l'Université Carnegie Mellon en 1985,
- Il est renommé Deep Thought en 1988,
- Il devient Deep Blue en 1993.

# Deep blue : quelques chiffres



- Deep Blue de 1996 mesurait 2m de haut et pesait 700 kg
- Il s'agissait d'un supercalculateur IBM (RS/6000 *Scalable POWER parallel Systems*) dont chacun des 32 processeurs consacrés au calcul pur était connecté à une carte comprenant 8 processeurs dédiés aux échecs, soit au total 256 processeurs spécialisés fonctionnant en parallèle.

# Deep blue

- Deep blue de 1996 mesurait 2m de haut et pesait 700 kg
- En 1997, Deep blue mesurait 1,80 m et pesait une tonne 4
  - Il fallait 20 personnes pour qu'il fonctionne
  - Deep Blue occupait la 259ème place au TOP 500 des supercalculateurs et avait une puissance de 11.38 GFLOPS.
- Deep Blue calculait alors entre 100 et 300 millions de coups par seconde, il pouvait calculer 12 demi-coup de profondeur en moyenne.
- Les grands maîtres d'échecs Miguel Illescas, John Fedorowicz, Nick De Firmian et Joel Benjamin aidèrent à sa conception, notamment pour sa bibliothèque d'ouverture.

# Les rencontres avec G Kasparov

- Deep Blue a rencontré le champion du monde d'échecs du moment, Garry Kasparov, le 10 février 1996. La machine remporte la première partie, mais Kasparov remporte trois autres parties et le match se solde par 4-2 en faveur du champion.
- En mai 1997 a lieu le match revanche en six parties entre Deep Blue et Kasparov. Le champion du monde dut s'incliner contre l'ordinateur, sur le score de 2,5 à 3,5

# Les parties de la rencontre de 1997

- Partie 1 : Kasparov gagne la partie facilement.
- Partie 2 : Kasparov abandonne prématurément après 45 coups, car après analyse il rata un coup menant à un échec perpétuel que la machine ne vit pas.
- Partie 3 : Kasparov assure une partie nulle.
- Partie 4 : partie nulle après 5 heures de jeu.
- Partie 5 : la partie était nulle par échec perpétuel.
- Partie 6 : Kasparov, fatigué physiquement et psychologiquement, perd la partie

# Après la confrontation

- Cette victoire a été contestée par Kasparov car l'équipe de programmeurs de Deep Blue avait accès à toutes les parties de Kasparov, tandis que celui-ci n'avait pas eu accès à la liste chronologique des parties jouées par Deep Blue.
- Une autre cause est à rechercher dans la stratégie de Kasparov, qui s'est écarté rapidement des sentiers battus de la théorie dans certaines parties, ce qui s'est retourné contre lui dans la dernière partie.
- Grâce à son incroyable force de calcul, Deep Blue a joué lors de ce match quelques coups de la classe d'un grand maître, dont l'un, abondamment commenté dans les revues spécialisées, a complètement déstabilisé Kasparov.



# Projet Hydra

- **Hydra** est un superordinateur dédié au jeu d'échecs
- Il a été conçu par l'équipe composée du Dr Christian « Chrilly » Donninger, d'Ulf Lorenz, du grand maître international Christopher Lutz et de Muhammad Nasir Al
- Depuis juillet 2005, Hydra est probablement l'entité la plus forte à ce jeu en confrontation temps réel.

# Hydra (1)

- Trois versions : Orthus, Chimera, Scylla
- Hydra est un ensemble de processeurs mis en grappe, interconnectés via fibre optique (*Myrinet 2000 Fiber 8-port* <sup>[1]</sup> ) et auxquels sont adjoints des circuits logiques programmables, tenant le rôle de coprocesseurs dédiés au jeu d'échecs.

# Hydra (2)

- Il peut évaluer 200 millions de positions par seconde
- Il est capable en moyenne d'évaluer jusqu'à 18 demi-coups (9 coups de chaque joueur) à l'avance
- Il utilise l'algorithme d'élagage alpha-beta, ainsi que l'optimisation heuristique à mouvement nul
- Hydra est physiquement situé à Abou Dabi, dans les Émirats arabes unis, et joue généralement au travers d'un réseau informatique

# Les résultats de Hydra

- En octobre 2004, Hydra a battu le champion du monde FIDE en titre, le russe Ruslan Ponomariov (niveau ELO de 2710) dans leurs deux rencontres.
- En juin 2005, Hydra a joué un match en six parties contre Michael Adams, le numéro 1 britannique et le numéro 7 au niveau mondial. Hydra a battu Adams par 5,5 points à 0,5
- En novembre 2005, Hydra a joué quatre parties contre trois ex-champions du monde, dans le cadre d'une rencontre hommes-machines.
  - Hydra a remporté une victoire contre Rustam Qosimjono,
  - Il a fait nulle contre Alexander Khalifman, puis
  - Il a remporté une autre victoire contre Ruslan Ponomariov et
  - Il obtenu une nulle contre Rustam Qosimjonov.

# Hydra : ses performances

- Hydra reste invaincu contre un humain non assisté en rencontre temps-réel.
- Hydra a été battue par des joueurs disposant d'accès à leurs ressources logicielles :

le grand maître international d'échecs par correspondance Arno Nickel a battu Hydra dans un match en deux parties qui a duré 6 mois

# Quelques chiffres

- Chess 3.0 pouvait évaluer environ 3600 coups par seconde
- Bell pouvait évaluer environ 150000 coups par seconde
- Deep Thought pouvait évaluer environ 700000 coups par seconde
- Deep blue calculait entre 100 et 300 millions de coups par seconde

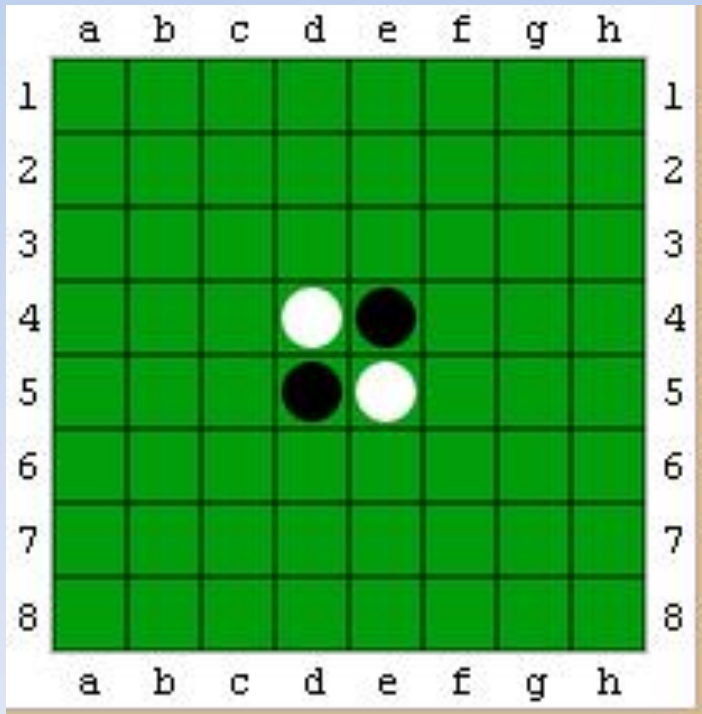
# Othello

# Othello

- « L'othellier » est un plateau de 8 cases sur 8, il doit être rempli de pions noirs sur une face et blancs sur l'autre
- Chaque joueur joue une couleur
- Le vainqueur est le joueur dont la couleur domine à la fin de la partie (en général quand toutes les cases sont jouées)
- Chaque coup consiste en le retournement d'au moins un pion adverse. Il faut pour cela encadrer un ou plusieurs pions de la couleur opposée entre un pion ami déjà présent et le pion joué lors de ce coup. Les pions adverses "encerclés" sur une ou plusieurs lignes sont alors retournés.



# Début du jeu



- Noir commence toujours et les deux adversaires jouent ensuite à tour de rôle.

# Historique succinct

- IAGO est écrit par l'université "Caltech" de Pasadena en 1976. Le champion japonais Fumio FUJITA le battit facilement.
- En juin 1980 est organisé le premier tournoi d'Othello® hommes-machines (3 micros et trois gros systèmes et 2 hommes).
- En janvier 1981 un tournoi de programmes regroupent 20 systèmes.
- Les années 1996-1997 voient exploser les systèmes.

# Les années 1990

- **Bill** réalisé par Kai-Fu Lee et Sanjoy Mahajan en 1988
- **Darwersi** réalisé par Olivier Arzac en 1996 (96-99)  
il utilise un algorithme génétique
- **Hannibal** réalisé par Martin Piotte et Louis Geoffroy en 1996
- **Keyano** réalisé par Mark Brockington en 97  
il utilise le parallélisme
- **Kitty** réalisé par Igor Durdanovic de NEC Research Institute (dans le New Jersey)
- **Logistello** réalisé par Michael Buro (pour sa thèse) de l'université de Paderborn (Allemagne) en 1997
- **Zebra** réalisé par Gunnar Andersson et Lars Ivansson en 2001

# Résultats obtenus en tournois

résultats obtenus de 1993 à 1997

Programme	Rang (1-2-3)	Auteurs
LOGISTELLO	18-6-0	<a href="#">M.Buro</a>
HANNIBAL	4-3-0	<a href="#">L.Geoffroy &amp; M.Piotte</a>
KITTY/REV	2-3-6	<a href="#">I.Đurđanovic</a>
ECLIPSE	2-2-2	<a href="#">M.Giles &amp; C.Springer</a>
KEYANO	1-3-2	<a href="#">M.Brockington</a>

# Comment joue le système ?

- Trois grandes étapes
- Le début de partie : création d'une base de donnée de débuts de parties
- Le milieu de partie : utilisation d'algorithmes de la famille  $\alpha\beta$
- La fin de partie : utilisation de l'algorithme  $\alpha\beta$  et de la méthode d'analyse rétrograde

# Optimisation des méthodes

- Trier les coups : plusieurs critères sont utilisés
  - Le type de case
  - Les coups qui minimisent la mobilité de l'adversaire
  - Faire une recherche à une profondeur courte
  - Privilégier les réponses habituelles au coup considéré
- Améliorer la fonction d'évaluation : elle utilise la structure des pions : par exemple l'othellier est divisé en zones, on attribue une note en fonction de la configuration des pions qui s'y trouvent

# Exemple de fonction d'évaluation grossière

Considérer les 3 critères suivants :

- le *matériel* : nombre de pions d'une couleur donnée
- la *mobilité* : nombre de cases jouables par cette couleur
- la *force* d'une position : somme des valeurs des cases occupées par cette couleur.

# Grille évaluant la valeur d'une case

500	-150	30	10	10	30	-150	500
-150	-250	0	0	0	0	-250	-150
30	0	1	2	2	1	0	30
10	0	2	16	16	2	0	10
10	0	2	16	16	2	0	10
30	0	1	2	2	1	0	30
-150	-250	0	0	0	0	-250	-150
500	-150	30	10	10	30	-150	500

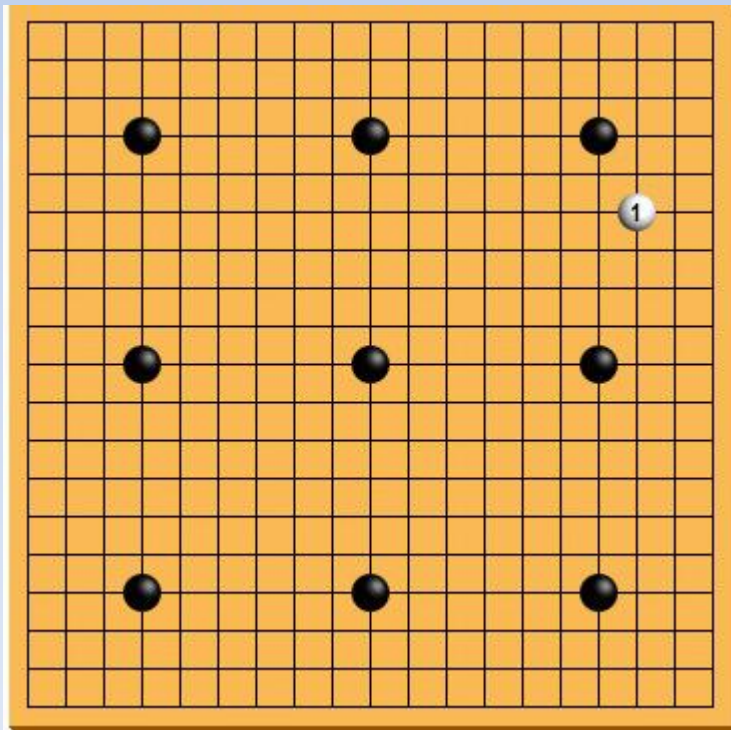


# Le jeu de Go

# Le go : règles de jeu

- A tour de rôle, les joueurs posent une pierre de leur couleur sur une intersection inoccupée du goban (19 x 19) ou bien ils passent.
- Lorsqu'un joueur supprime la dernière liberté d'une chaîne adverse, il la capture en retirant du goban les pierres de cette chaîne.
- Un joueur, en posant une pierre, ne doit pas redonner au goban un état identique à l'un de ceux qu'il lui avait déjà donné.
- La partie s'arrête lorsque les deux joueurs passent consécutivement. On compte alors les points. Chaque intersection du territoire d'un joueur lui rapporte un point, ainsi que chacune de ses pierres encore présentes sur le goban.

# Exemple : début de partie avec un handicap



Le nombre de possibilités est immense, il est estimé à  $10^{600}$  pour un goban 19x19, soit plus que le nombre de particules de l'Univers

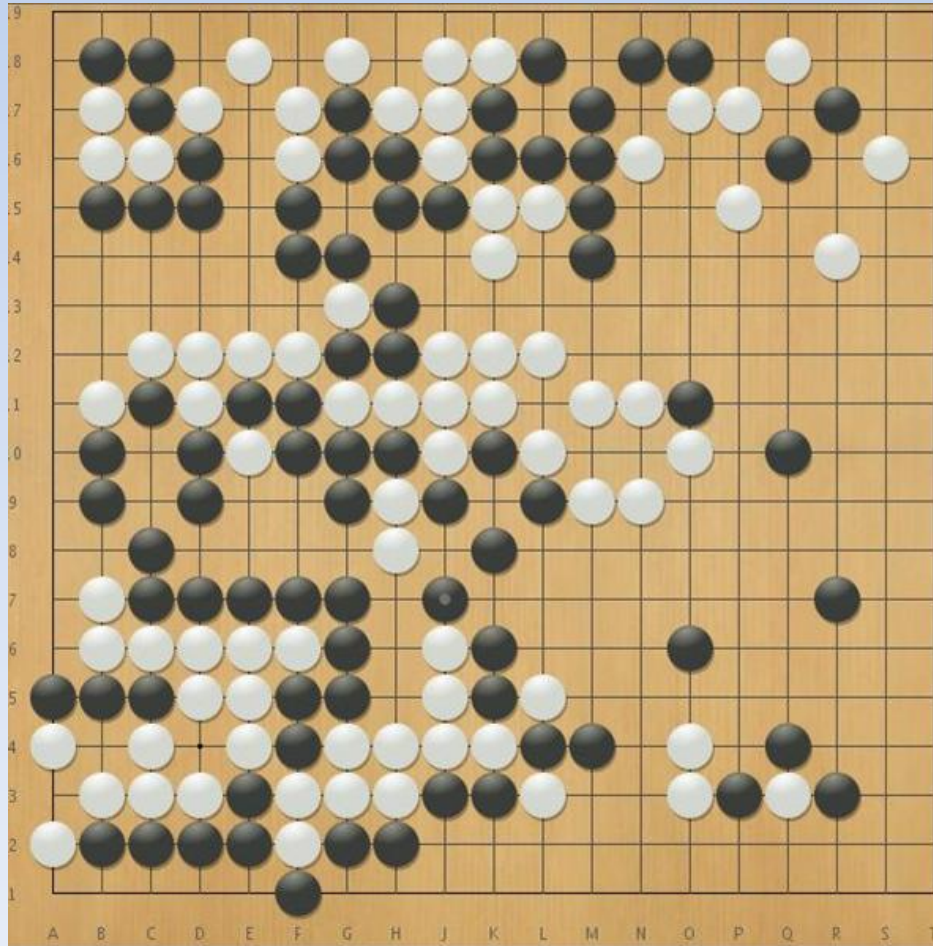
# Le jeu de Go

- Le premier programme a été écrit en 1968 par Albert Zobrit comme un élément de sa thèse sur la reconnaissance des formes.
- En 1997, Janice Kim, shodan professionnelle, battait le programme HandTalk malgré un handicap de 25 pierres
- À partir de 2006 les programmes parviennent désormais à égaler des joueurs de haut niveau sur un goban de taille 9x9 ou à des handicaps de 6 à 9 pierres sur un goban de taille 19x19

# MoGo

- MoGo, classé numéro 1 depuis août 2006 parmi 142 sur le Computer Go Server (tournois de matchs en 9x9), a été développé au sein du projet TAO en collaboration avec le CMAP de l'Ecole Polytechnique.
- Dans les tournois entre ordinateurs sur KGS, MoGo a gagné dans toutes les tailles: 9x9, 13x13, 19x19
- Cette équipe est formée notamment d'un
  - doctorant à l'Inria, Sylvain Gelly et
  - d'un ancien joueur de Go du COP, en master au CMAP (centre de Mathématiques appliquées) de l'école Polytechnique Yizao Wang.

# La première partie jamais gagnée à handicap 7 contre un joueur professionnel 9<sup>e</sup> dan Pro



# Les nouveautés de MoGo

Il s'attaque surtout aux deux problématiques suivantes:

- **Evaluation de la position par Monte-Carlo** : à partir d'une position on fait jouer le programme au hasard jusqu'au bout de la partie. En effectuant cette opération de nombreuses fois, et moyennant les résultats, nous obtenons alors une évaluation de la position.
- **Exploration-exploitation dans l'arbre de recherche par l'algorithme UCT** : il consiste à choisir de ré-évaluer un coup avec une probabilité d'autant plus forte que sa valeur évaluée actuellement est grande.

# Méthode de Monte Carlo

- Méthode visant à calculer une valeur numérique, et utilisant des procédés aléatoires, c'est-à-dire des techniques probabilistes.
- Elle a été inventé en 1947 par Nicholas Metropolis, et publié pour la première fois en 1949 dans un article co-écrit avec Stanislas Ulam.



# Idées essentielles de Monte Carlo

- Ne pas choisir les coups de manière uniformément aléatoire. Certains coups qui semblent meilleurs a priori (comme répondre à une menace immédiate) peuvent être choisis avec une probabilité plus élevée, ce qui améliore la précision de l'évaluation.
- Augmenter la probabilité de choisir un coup qui s'est avéré être bon dans les simulations précédentes.

# Crazy stone

- Il est développé par Remi Coulom, un informaticien français, depuis 2005.
- L'algorithme de Monte-Carlo est incorporé à sa conception
- La reconnaissance de combinaisons et la recherche sont ajoutées en 2006

# Résultats récents

- 2008 : Crazy Stone a battu Kaori Aoba (4<sup>e</sup> dan, professionnel) deux fois avec un handicap de 8 pierres, puis avec un handicap de 7 pierres,
- 2012 : Le programme Zen a battu Takemiya Maski (9<sup>e</sup> dan) deux fois, avec un handicap de 5 et 4 pierres,
- 2013 : Crazy Stone a battu Ishida Yoshio (9<sup>e</sup> dan) avec un handicap de 4 pierres.

# AlphaGo

- AlphaGo a été développé par DeepMind, une entreprise britannique spécialisée dans l'intelligence artificielle créée par Demis Hassabis, Mustafa Suleyman et Shane Legg en 2010 et rachetée par Google en 2014.
- AlphaGo représente une amélioration significative par rapport aux précédents programmes de Go. Sur 500 parties jouées contre d'autres programmes, y compris Crazy Stone et Zen, AlphaGo n'en a perdu qu'une.

# AlphaGo

AlphaGo repose sur trois « techniques »

- Monte-Carlo
- Réseaux de neurones profonds (deep learning)
- Apprentissage renforcé (faire jouer AlphaGo contre lui-même)

# AlphaGo : Deep learning

Deux réseaux

- *Policy network* permet de prédire le prochain coup
- *Value network* estime l'issue favorable d'un mouvement à partir de la configuration du plateau de jeu.

Chaque réseau est composé de 12 couches contenant chacune des millions de connexions neuronales

# AlphaGo : résultats

- Octobre 2015 : AlphaGo bat le champion européen FanHui (2<sup>e</sup> dan) sur un goban de taille normale (19x19)
  - 5 victoires à 0 en parties lentes
  - 3 victoires à 2 en parties rapides
- Mars 2016 : AlphaGo a battu le sud-coréen Lee Sedol (9<sup>e</sup> dan professionnel, niveau maximum), par 4 parties contre 1

# Le match de mars 2016

Le match en 5 parties se termine par la victoire 4-1 de l'ordinateur :

- Victoire d'AlphaGo avec les blancs.
- Victoire d'AlphaGo avec les noirs.
- Victoire d'AlphaGo avec les blancs.
- Victoire de Lee Sedol avec les blancs en offrant un jeu de qualité face à quelques preuves de faiblesse d'AlphaGo.
- Victoire d'AlphaGo avec les blancs. Lee Sedol joue les noirs à sa demande considérant qu'ainsi la partie aurait plus de valeur.



# Conséquences

- Avant même la dernière partie et assuré de la victoire, AlphaGo est classé 9<sup>e</sup> dan professionnel de manière honorifique par la Hanguk Kiwon (fédération coréenne de jeu de go).
- Le prix d'un million de dollars, gagné par l'équipe d'AlphaGo est donné à des associations caritatives.

# Conclusion

Jeu	Complexité	État de l'art
Morpion	$10^3$	Résolu (à la main) : partie nulle si les joueurs jouent bien
Puissance 4	$10^{14}$	Résolu en 1988 : le joueur qui commence a une stratégie gagnante
Dames anglaises	$10^{20}$	Résolu en 2007 : partie nulle si les joueurs jouent bien
Échecs	$10^{50}$	Non résolu. Les programmes sont meilleurs que les meilleurs humains
Go	$10^{171}$	Non résolu. Les programmes se rapprochent des meilleurs