

Datastructuren project voor KI

- Docent: **José Lagerberg**
- Assistenten: **Auge Wiggers** en **Tim van Rossum**
- Cursusinformatie: <http://www.science.uva.nl/~jose/>
- Project: grote programmeeropgave in Java

Toetsing van project

Cijfer van project is bepaald door:

- 1 correcte implementatie en juiste uitvoering van programma
- 2 leesbaarheid en layout van ingeleverde code
- 3 structurele opbouw van programma
- 4 exception handling
- 5 logboek van gedane werkzaamheden
- 6 eindverslag van ongeveer 2 pagina's
- 7 gebruik KI algoritmen
- 8 originaliteit
- 9 demo van uiteindelijke programma

Eindcijfer Datastructuren is $P/6 + T/3 + \text{project}/2$

Programmeeropgave in Java

- Programmeeropgave in tweetallen
- **Verplichte aanwezigheid** op practicum op elke woensdag
- Elk team moet **elke week** voortgang met docent/assistent bespreken
- Logboek moet bijgehouden worden voor al het gedane werk
- **Inleveren** in woensdag 19 maart 23.00
- **Demo** in donderdag 20 maart gegeven door tweetal
- Het zonder bronvermelding **copiëren** van code van internet wordt gezien als **fraude**

Opgaven

Er kan een keuze gemaakt worden uit één van de volgende spelletjes:

- Sudoku-oplosser
- Othello
- Vier-op-een-rij
- Andere spelletjes zijn misschien ook mogelijk

Sudoku-oplosser

- Het programma moet de in te lezen sudoku oplossen met strategische stappen (dus geen backtracking)
- Hierbij is **OOP** aanpak belangrijk
- Van te voren moeten klassen en methoden gespecificeerd worden
- Oplosstappen als **single**, **hidden pair**, **locked pair** moeten geïmplementeerd worden

Othello

- Hierbij speelt de computer tegen een **tegenstander**
- Dat betekent dat de beste zet voor de computer bepaald moet worden.
- Dat gaat vaak met **recursie**
- Gebruik van **minimax**-algoritme

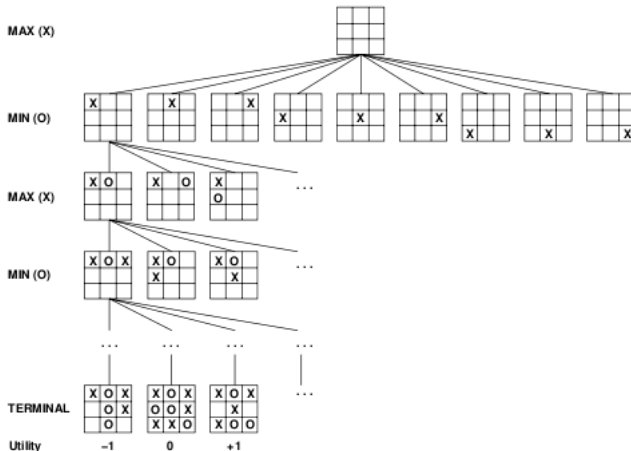
Vier-op-een-rij

- Hierbij speelt de computer tegen een **tegenstander**
- Dat betekent dat de beste zet voor de computer bepaald moet worden.
- Dat gaat vaak met **recursie**
- Gebruik van **minimax**-algoritme

Denkspelen

- Programma's zoals schaak, othello, of vier-op-een-rij, die van u proberen te winnen hebben allemaal eenzelfde structuur
- Een zoekprocedure onderzoekt alle mogelijke vervolgen van het spel
- Al de mogelijke spelontwikkelingen kunnen schematisch worden weergegeven met een boomstructuur: **spelboom**
- **bladeren** van de spelboom zijn die opstellingen waarin het spel geëindigd is
- spelboom van meeste interessante spellen is te groot, daarom doorzoeken van spelboom tot op zekere **maximale diepte**
- zoeken dan afgebroken bij stelling waarvan computer nog niet kan zeggen of hij **gewonnen** is of **verloren**
- Aan dergelijke stelling wordt dan **score** toegekend

Spelboom van boter-kaas-en-eieren



Eerste practicum project op woensdag 19 februari

- 1 Iedereen moet partner gekozen hebben
- 2 Verder moet keuze uit 3 spelletjes gemaakt zijn
- 3 Woensdag 19 februari moet keuze met docent/assistent besproken worden

Het project wordt volgens dit milestone-schema uitgevoerd:

- 1 donderdag 20 februari: projectvoorstel
- 2 woensdag 26 februari: ontwerpdocument
- 3 woensdag 5 maart: alfaversie
- 4 woensdag 19 maart voor 23.00 uur: releaseversie
- 5 donderdag 20 maart: demos