

Inverse Kinematica/HighPath opdracht

- De huidige simulator moet kunnen werken met een tekstfile van hoeken/robotparameters als input
- Een Python file schrijven die als input een tekstbestand (of jsonfile) van x,y,z coördinaten heeft en als output aan tekstbestand aan robotparameters/hoeken geeft (dit wordt het framework voor de inverse kinematicaopdracht)
- Een Python file schrijven die als input een bordpositie en een move (bijv. d6d8) meeneemt en als output de x,y,z coördinaten van de highpath route teruggeeft
- (Eventueel) ervoor zorgen dat de outputfile van robotparameters ook daadwerkelijk kan worden gebruikt bij de echte robot

Chess engine

- Testcases ontwikkelen
- Evaluatiefunctie interessanter maken
- Een tekstdocument met de opdrachtbeschrijving maken
- Het framework afmaken (de code die ze zelf moeten invullen eruit halen, commentaar toevoegen om duidelijk te maken wat ze moeten doen, etc.)

Robots voor projecten

- Mini-drones handleiding afmaken
- "Nieuwe robot" aan de praat krijgen
- AR-drones repareren
- Een introductie voor OpenCV en ROS maken
- Ideeën voor niet-robotgerelateerde problemen uitwerken

Robots:

AR-drones: max 8 groepjes -> relatief makkelijk om mee te beginnen

=> moeten nog gerepareerd worden

Icat: max 2 groepjes -> moeilijk om in te komen, want c++

Nao's: max 10 groepjes -> gemiddeld

Pepper: max 2 groepjes -> gemiddeld, iets moeilijker omdat er geen voorbeeldcode is

Mini-drones: max 5 groepjes -

Assembleer <http://scanse.io/3d-scanning-kit/>

Overig:

Opencv

ROS

Schaken verbeteren -> quiescent search

AR-drones: 8 (maar er werken er een paar niet, eventueel twee groepjes 1 drone)

Icat: 2 (zoveel als maar willen, er is een simulator)

Nao's: 5-6 nao's, dus 10 groepjes

Pepper: 1 of 2 groepen

