**Practicum TIR KIN 02**

Vervaardig één van de volgende producten:

1. **Programma dat wiskundige formules manipuleert**
   1. Ontwerp een LL1 syntax voor formules met tenminste: optellen, aftrekken, vermenigvuldigen, delen, machtsverheffen
   2. Vervaardig handmatig een recursive descent parser voor deze syntax die een visuele syntaxtree genereert in de vorm van een geindenteerde lijst
   3. Vervaardig een character recogniser in de vorm van een neural network met backpropagation dat de betreffende syntax van gestyleerd handschrift omzet in ASCII strings
   4. Vervaardig een rule based inference system dat formules die zich daarvoor lenen omzet in standaard polynoom vorm, bijv. ax^3 + bx^2 + cx + d. Met deling hoeft geen rekening te worden gehouden
   5. Breidt dit systeem uit met deling en als eindresultaat het quotient van twee polynomen
2. **Programma dat muziek genereert**
   1. Schrijf een functie die willekeurige groep tonen simultaan kan laten klinken
   2. Bedenk een ASCII syntax voor representatie van een polyfone muziekstukken inclusief tempo, duur van de noten en rusten
   3. Maak een muziekgenerator die random muziekstukken en variaties daarop genereert en opslaat in genoemde syntax
   4. Maak rating door gebruikers mogelijk en werk via een genetisch algoritme de muziekstukken met de hoogste rating verder uit
   5. Breng heuristiek in: eenvoudige frequentieverhoudingen, vaste maatindeling, II V I accoord sequence, slotnoot = toonsoort, laddereigen noten etc.
3. **Programma dat stratego of schaak speelt tegen menselijke tegenspeler**
   1. Vervaardig een userinterface dat twee menselijke spelers in staat stelt tegen elkaar te spelen
   2. Breidt dit uit met correctheidscontrole van de zetten
   3. Laat de computer vanuit een random gegenereerd beginsituatie van zijn stukken steeds een willekeurige toegestane zet doen als antwoord op een menselijke zet
   4. Laat de computer zetten doen op basis van resultaatevaluatie na een willekeurig aantal ply's, met als enige goal het verlies van zo min mogelijk stukkenwaarde (uitgedrukt in punten)
   5. Laat de computer een verstandige beginsituatie genereren en bij elke zet tevens heuristiek meenemen (vlag tussen de bommen, spion naast maarschalk, openingen, eindspelen)

Toegestane talen: Python, C++, C#, Objective C, Java, JavaScript

De volgende algemene library categorieen zijn toegestaan: Vector/matrix/numerical , GUI algemeen, OpenGL, Midi drivers, Boost, Swig

Niet toegestaan zijn specifieke libraries gericht op bovenstaande toepassingen, waaronder libraries voor: game development, neural nets, rule based systems, parsers en parsergenerators, character recognition, symbolische algebra, muziekgeneratie

Bij twijfel of gebruik van een bepaalde library is toegestaan overleggen met docent.