

Nascholing

Algoritmisch Denken



Martin Bruggink
Renske Smetsers





Petals Around the Rose



Tip: bloemblaadjes rond de roos

Petals Around the Rose



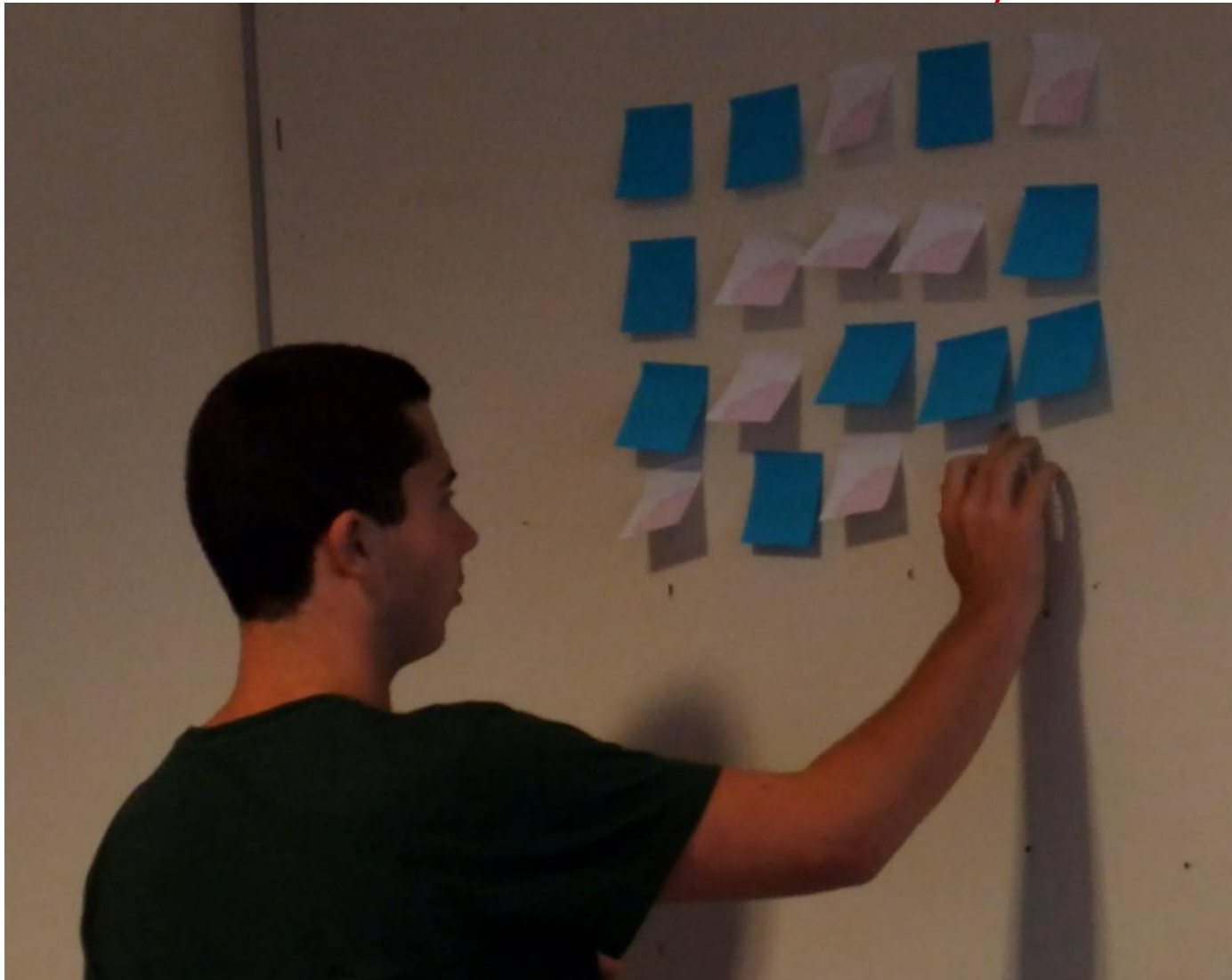
Tip: bloemblaadjes rond de roos: **ABSTRACTIE!!**

Speerpunten bijeenkomst #3

- Afrondend bijeenkomst
- Terugkoppeling & reflectie
 - Sorteren
 - BitParity
- Verdere kennismaking met Alg. Denken
 - Puzzel opdracht
 - Greenfoot lesmateriaal
- Over automaten
- Wat verder ter tafel komt

Ervaringen mbt sorteren / muddy
city

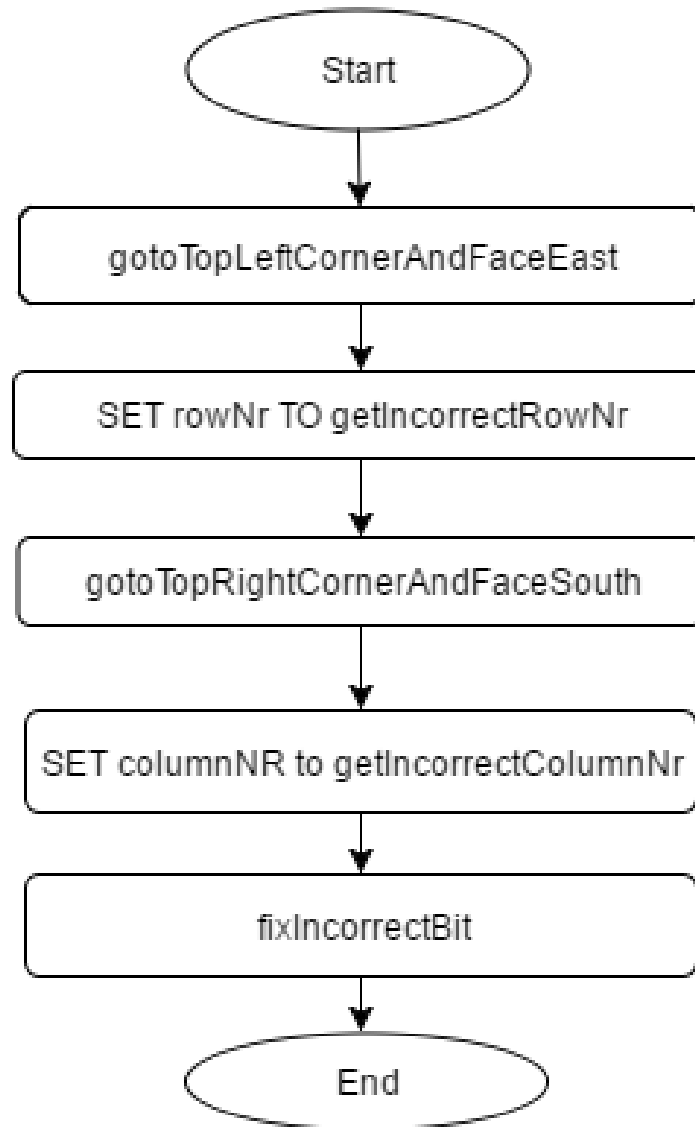
Reflectie BitParity



BitParity and the big five

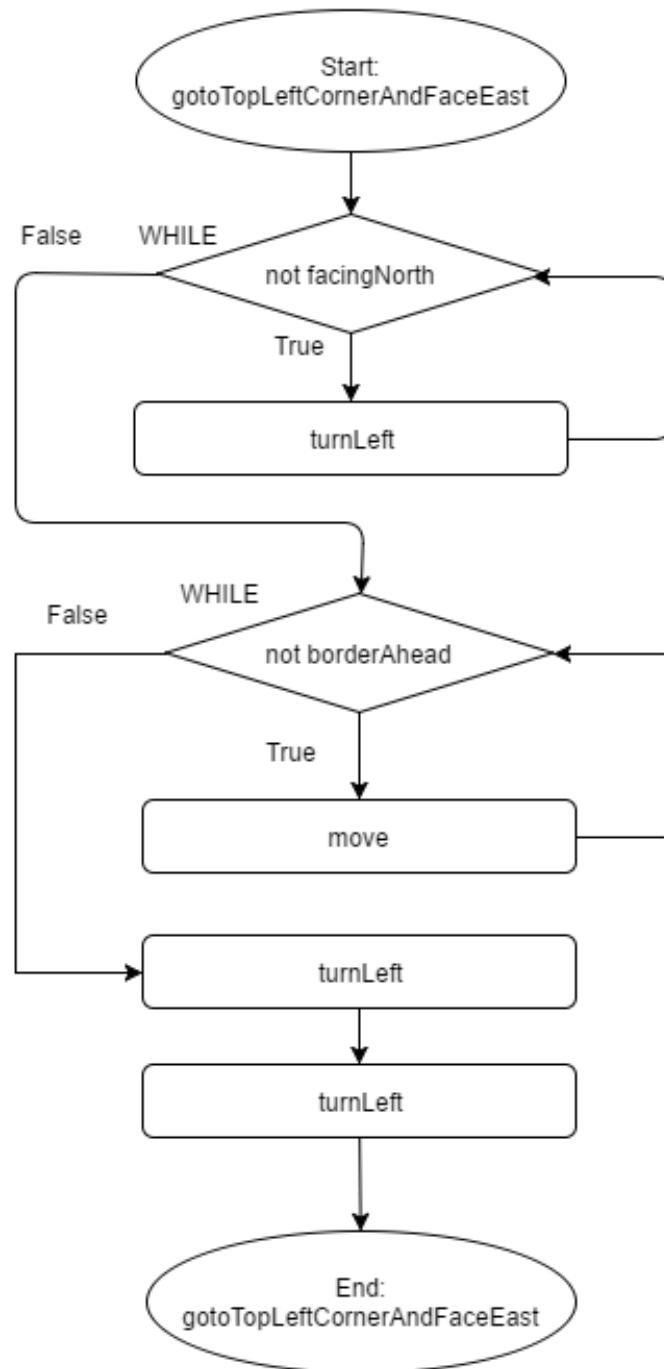


Rhino deel 2:



Beginsituatie: Dodo op een willekeurige positie in wereld

Rhino deel 2:



BitParity & the big five

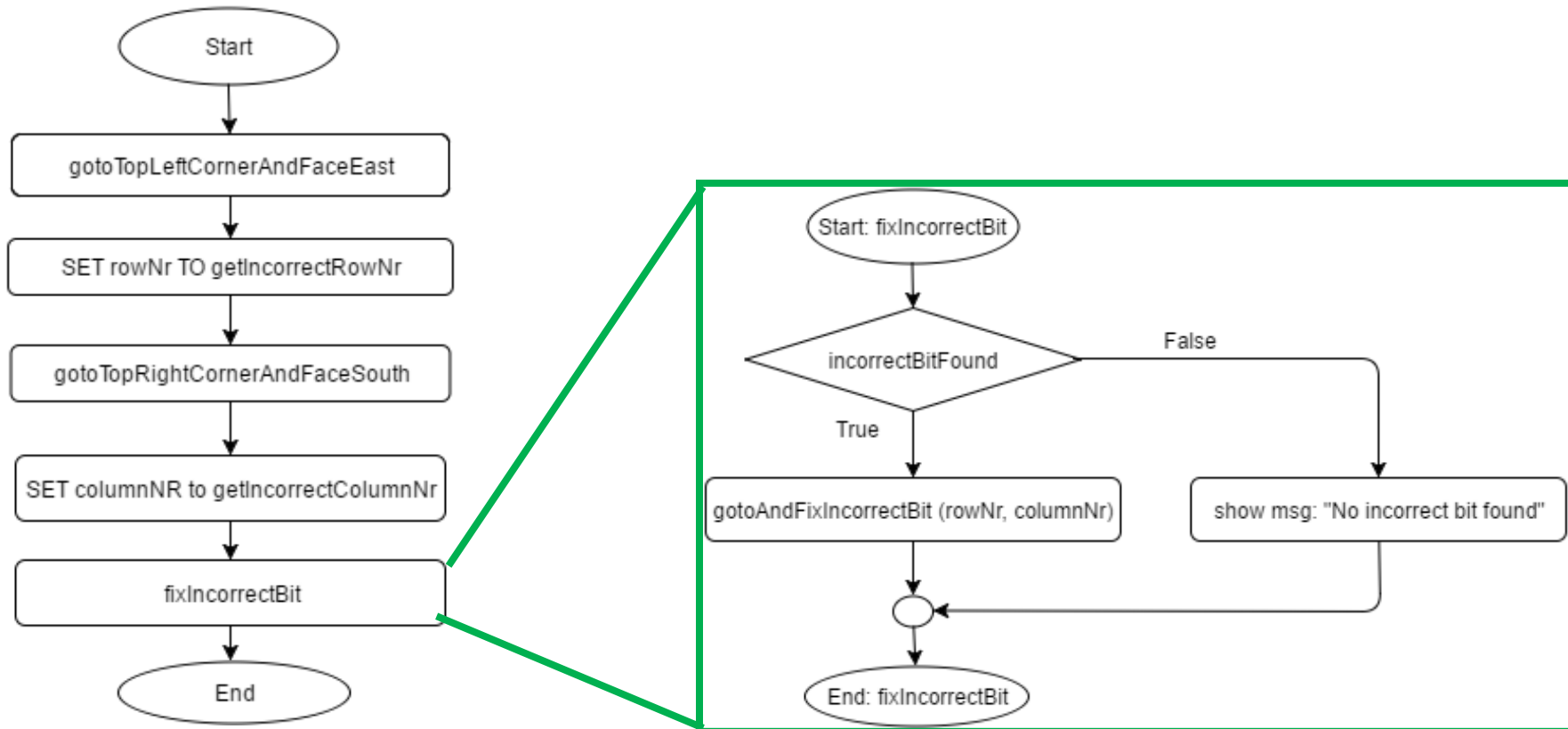
- Algoritmiek: denken in regels en opeenvolgingen (van deeltaken)
- Abstractie: weglaten van details
- Decompositie: opdelen in deelproblemen
- Generalisatie: patronen & generieke opl.
- Evaluatie: voldoet oplossing aan eisen?
had het beter/makkelijker gekund?

BitParity: Algoritmiek

- Conditional statements
 - `not facingNorth`, `not borderAhead`
- Boolean statements
 - `moreRowsToCheck`, `isEven`
- Bedenk Pre- en post condities:
 - `countEggsInRow`: gaat na afloop terug naar begin van rij

Bitparity: Abstractie

- Algoritmes op hoog-niveau



BitParity: Decompositie

Methodes en submethods

- `fixIncorrectBit()`
 - Check of ei gelegd of weggehaald moet worden en repareert deze
- `countEggsInRow()`
 - Tel aantal eieren in rij

BitParity: Generalisatie

- Herken mogelijkheden voor hergebruik
- Voordelen en nadelen van hergebruik afwegen
- Minder afhankelijk van details:
 - Wereld grootte
 - Richtingsgevoel

BitParity: Generalisatie

- Mogelijkheden voor hergebruik: `countEggsInRow`
- Hoe generiek is `GoToInitialPosition`?
Keuze voor relatieve implementatie maakt oplossing eenvoudiger, bv `loopNaarHoekLinksAchterJe`:
 - Geen noodzaak richtingsgevoel
 - Hergebruik `getIncorrectRowNr` ook voor kolom
 - Keuze is bepalend voor verdere keuzes die je maakt, bv hoe kolommen aflopen en `gotoIncorrectBit`

BitParity: Evaluatie

- Correctheid?
 - Werkt ook bij 0 of 2 incorrect bits?
- Herbruikbaar?
 - Submethodes, JavaDoc, Pre-post condities
- Mogelijke verbeteringen:
 - Oplossing
 - Aanpak

BitParity Samenvattend

The big five:

- Algoritmie: denken in regels en opeenvolgingen (van deeltaken)
- Abstractie: weglaten van details
- Decompositie: opdelen in deelproblemen
- Generalisatie: patronen & generieke opl.
- Evaluatie: voldoet oplossing aan eisen?
had het beter/makkelijker gekund?

Nieuw examenprogramma, grondslagen

Algoritmen

De kandidaat kan

- een oplossingsrichting voor een probleem uitwerken tot een algoritme;
- daarbij standaardalgoritmen herkennen en gebruiken;
- de correctheid en efficiëntie van digitale artefacten onderzoeken via de achterliggende algoritmen.

Datastructuren

De kandidaat kan verschillende abstracte datastructuren met elkaar vergelijken op elegantie en efficiëntie.

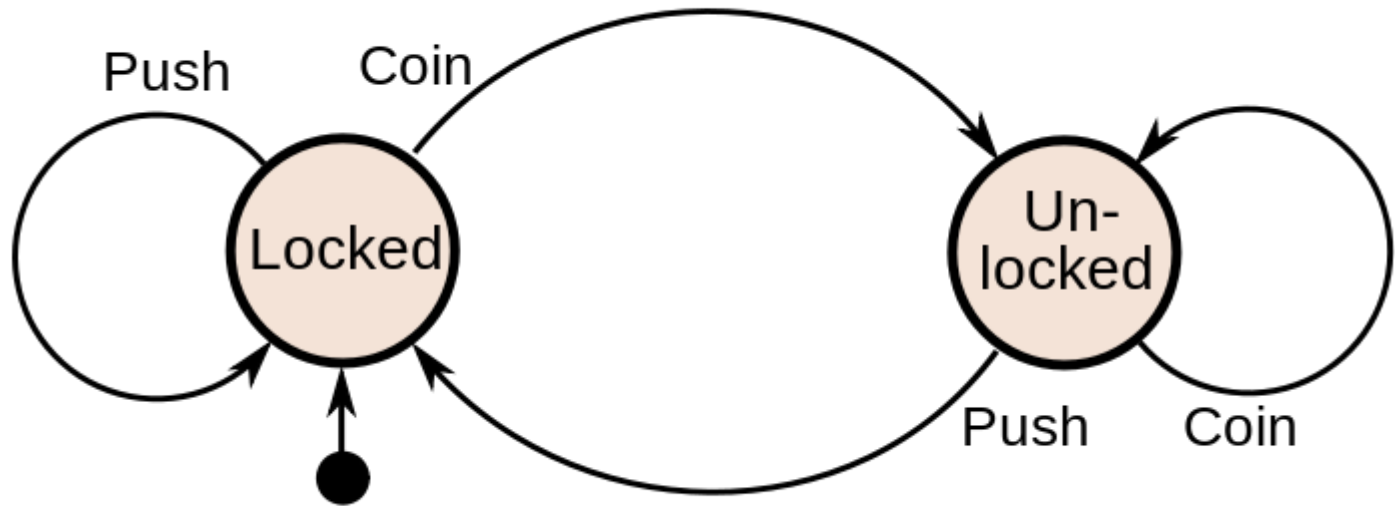
Automaten

De kandidaat kan eindige automaten gebruiken voor de karakterisering van bepaalde algoritmen.

Grammatica's

De kandidaat kan grammatica's hanteren als hulpmiddel bij de beschrijving van talen.

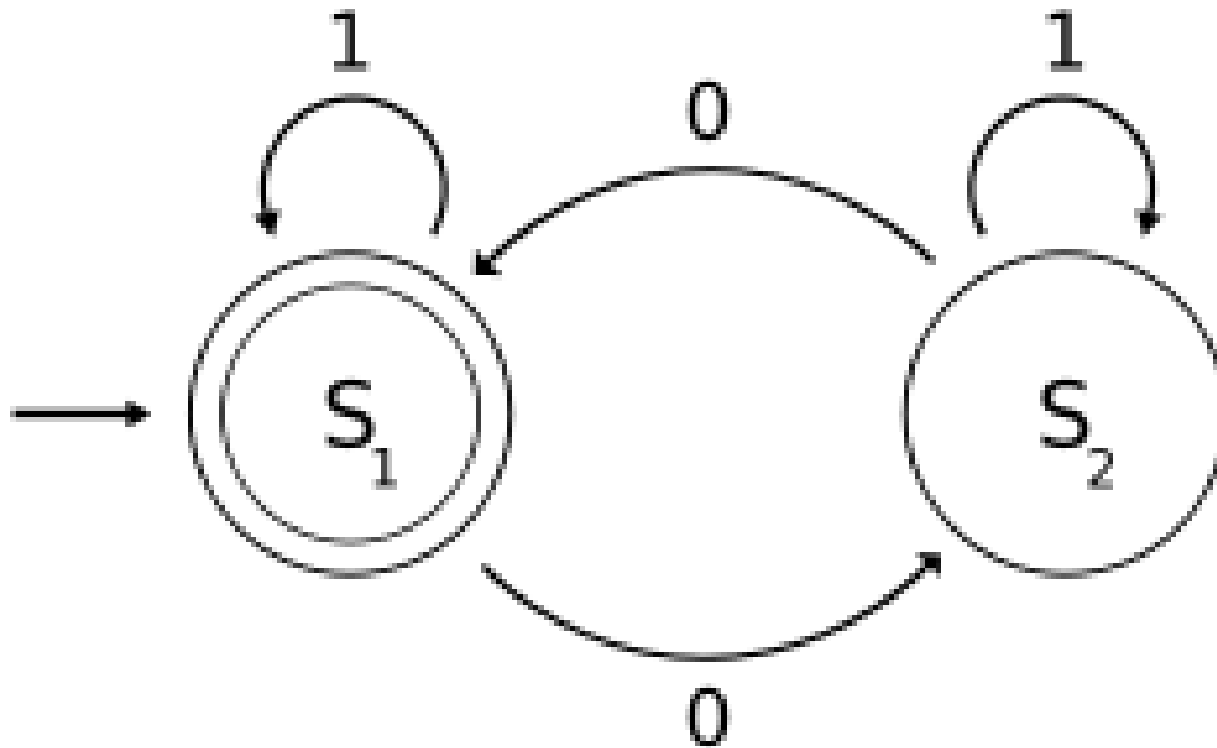
Automaten



Automaten



Automaten: voorbeeld



Automaten: model-checking

The screenshot shows a web browser window displaying a Moodle course page. The browser's address bar shows the URL <https://moodle.informatica-actief.nl/course/view.php?id=444>. The page header includes the course name 'INF-Actief', the language 'Nederlands (nl)', and the user 'Martin Bruggink'. The breadcrumb trail is 'Home > Blok 5: Projectmatig ontwikkelen > B5 Model Checking'. A left-hand navigation menu is visible, listing various course sections under 'B5 Model Checking'. The main content area is titled 'Model Checking' and contains two paragraphs of text. The first paragraph discusses the increasing complexity of computers and the need for model checking to find software bugs. The second paragraph explains the trend of using models in system development, mentioning UML and the Uppaal model checker. To the right of the text is a graphic for 'UPPAAL 4.0' featuring a state transition diagram and logos for Uppsala University and Aalborg University. At the bottom of the browser window, there are icons for 'Woordenlijst Model Checking' and 'Mededelingen'. The Windows taskbar at the very bottom shows the system clock as 21:49 on 6-12-2016.

NAVIGATIE

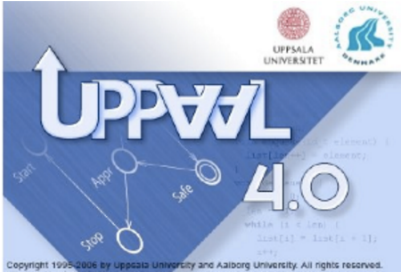
- Home
 - Mijn startpagina
- INF-Actief
- Huidige cursus
 - B5 Model Checking**
 - Deelnemers
 - Badges
 - Algemeen
 - 1: Wat is Model Checking?
 - 2: Inleiding in Uppaal
 - 3: Tijd en Klokken
 - 4: Overige mogelijkheden van Uppaal
 - 5: Gedistribueerde algoritmen
 - 1: Inleiding
 - 6: Gedistribueerde algoritmen
 - 2: het alternating b...
 - 7: Gedistribueerde algoritmen
 - 3: het kiezen van ee...
 - Mijn cursussen

Model Checking

Computers rukken steeds verder op. Vliegtuigen, treinen en auto's zitten vol met electronica. Soms gaat het mis: dan komt een vliegtuig in een duikvlucht omdat de computer de macht overneemt, of stopt een auto midden op de snelweg vanwege een software bug. Hoe kunnen we dit soort fouten voorkomen? Om ervoor te zorgen dat software bugs al tijdens het ontwerp worden gevonden (en niet pas wanneer bijvoorbeeld het vliegtuig al in de lucht is) maken bedrijven steeds vaker gebruik van modellen. Deze modellen bestaan uit diagrammen en tekst waarmee op een hoog niveau van abstractie wordt beschreven hoe cruciale onderdelen van een computersysteem zich gedragen. Vervolgens gebruikt men slimme software om aan die modellen te rekenen: men probeert fouten op te sporen of te bewijzen dat er geen fouten zijn.

Binnen de informatica wordt modelleren steeds belangrijker. Er is sprake van een duidelijke trend waarbij modellen het voornaamste product van systeemontwikkeling worden, denk bijvoorbeeld aan de opkomst van de Unified Modeling Language (UML). Modellen kunnen worden gebruikt voor overleg met klanten en belanghebbenden, voor het analyseren van het ontwerp, voor het genereren van code en voor het testen van de uiteindelijke implementatie.

Tijdens deze cursus werk je met het softwarepakket Uppaal, een zogenaamde model checker waarmee je zelf modellen kunt maken en laten doorrekenen. Je gebruikt Uppaal om een aantal systemen te modelleren en analyseren. Zo ga je op zoek naar fouten in het ontwerp van een geldautomaat, een blikjesautomaat, en een spoorwegovergang, en probeer je deze fouten te repareren. Ook leer je hoe je een model checker kunt gebruiken voor het oplossen van puzzels en denkspellen zoals "De wolf, de geit en de kool" en "Rush hour". Er zijn ook een paar meer geavanceerde facultatieve onderwerpen zoals algoritmen voor mutual exclusion, gedistribueerde algoritmen voor het kiezen van een leider in een ring, en het alternerend bit protocol (een vereenvoudigde versie van een algoritme dat gebruikt wordt binnen het internet). Doel van de cursus is dat je leert werken met een model checker, in het bijzonder aanpassingen kunt maken aan bestaande modellen, zelf eenvoudige modellen kunt construeren, en gewenste eigenschappen kunt formuleren en uitrekenen met behulp van een model checker.



UPPSALA UNIVERSITET
AALBORG UNIVERSITY

Copyright 1995-2006 by Uppsala University and Aalborg University. All rights reserved.

Woordenlijst Model Checking

Mededelingen

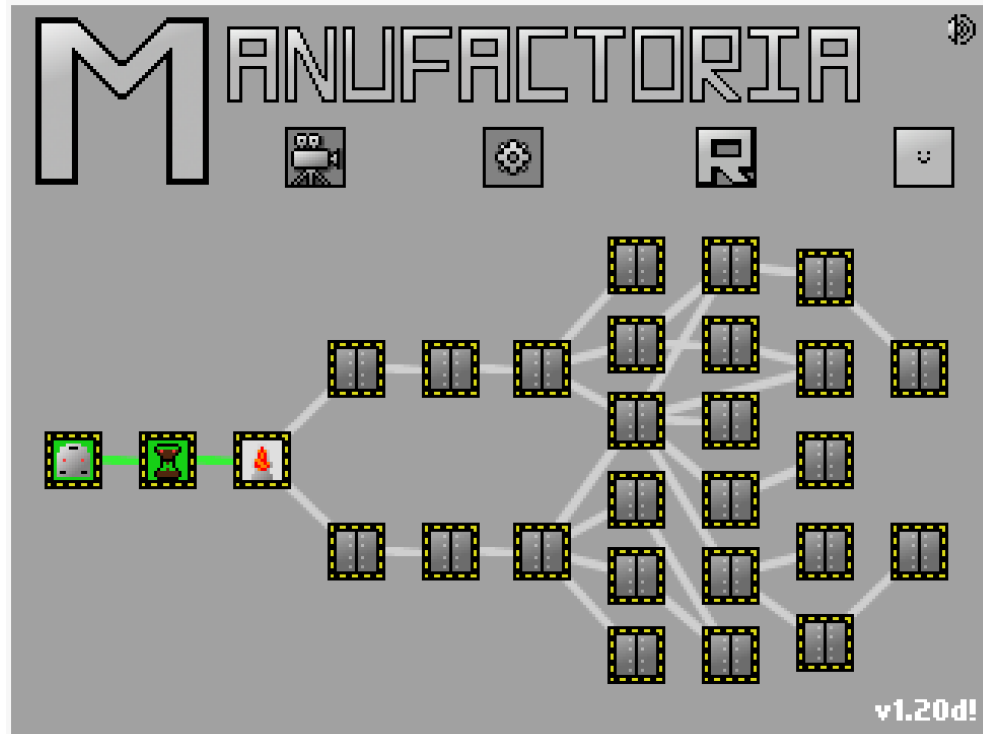
21:49
6-12-2016

Automaten: wolf, kool, geit

The screenshot shows a web browser window with several tabs open. The active tab is 'Plastelina Logic Games' and the address bar shows 'www.plastelina.net/game1.html'. The page content includes a sidebar with a list of logic games and a main game area. The sidebar lists: Wolf Sheep & Cabbage, Cannibals & Missionaries, Family Crisis, The Lonely Knight, Elevators Logic, Bags & Signs, Frogs Logic, Queens Logic, Knights Logic, Scales Logic, Golo's Dinner, Golo In The Cave, Logic Matches, Break The Machine 1, Break The Machine 2, and To be continued... The main game area features a 'PLAY' button, a game illustration of a man in a boat on a lake with a wolf, sheep, and cabbage on the shore, and a text box with instructions: 'please help the man in the boat to move - the wolf , the sheep and the box of cabbage to the other side of the lake. notice that: wolves eat sheep & sheep eat cabbage when no man around.' Below the text box is a 'NEXT' button and a text input field containing 'W o l f s h e e p & c a b b a g e' with a 'M R' button to its right. The Windows taskbar at the bottom shows the time as 22:08 on 6-12-2016.

<http://www.plastelina.net/game1.html>

Automaten: spel



<http://jayisgames.com/games/manufactoria/>

Automaten: bronnen

- CS Unplugged: Treasure Island / Finite State Automata (<http://csunplugged.org/finite-state-automata/>)
 - Bijbehorende Greenfoot:
<http://greenroom.greenfoot.org/resources/5>
- <http://www.swisseduc.ch/compscience/karatojava/kara/>
- <http://dl.acm.org/citation.cfm?id=2940331> (artikel over leren over automata)

AD diagonals

Uitdaging: Tour Guide

- Bedenk een oplossing:
 - Start & eindig bij hotel
 - Bezoek elk locatie precies 1 keer
- Noteer jouw algoritme / route
 - mbv pijltjes of nummertjes
 - Zodat jouw schoonzus het volgend weekend ook kan uitvoeren
- Evalueer of oplossing aan eisen voldoet

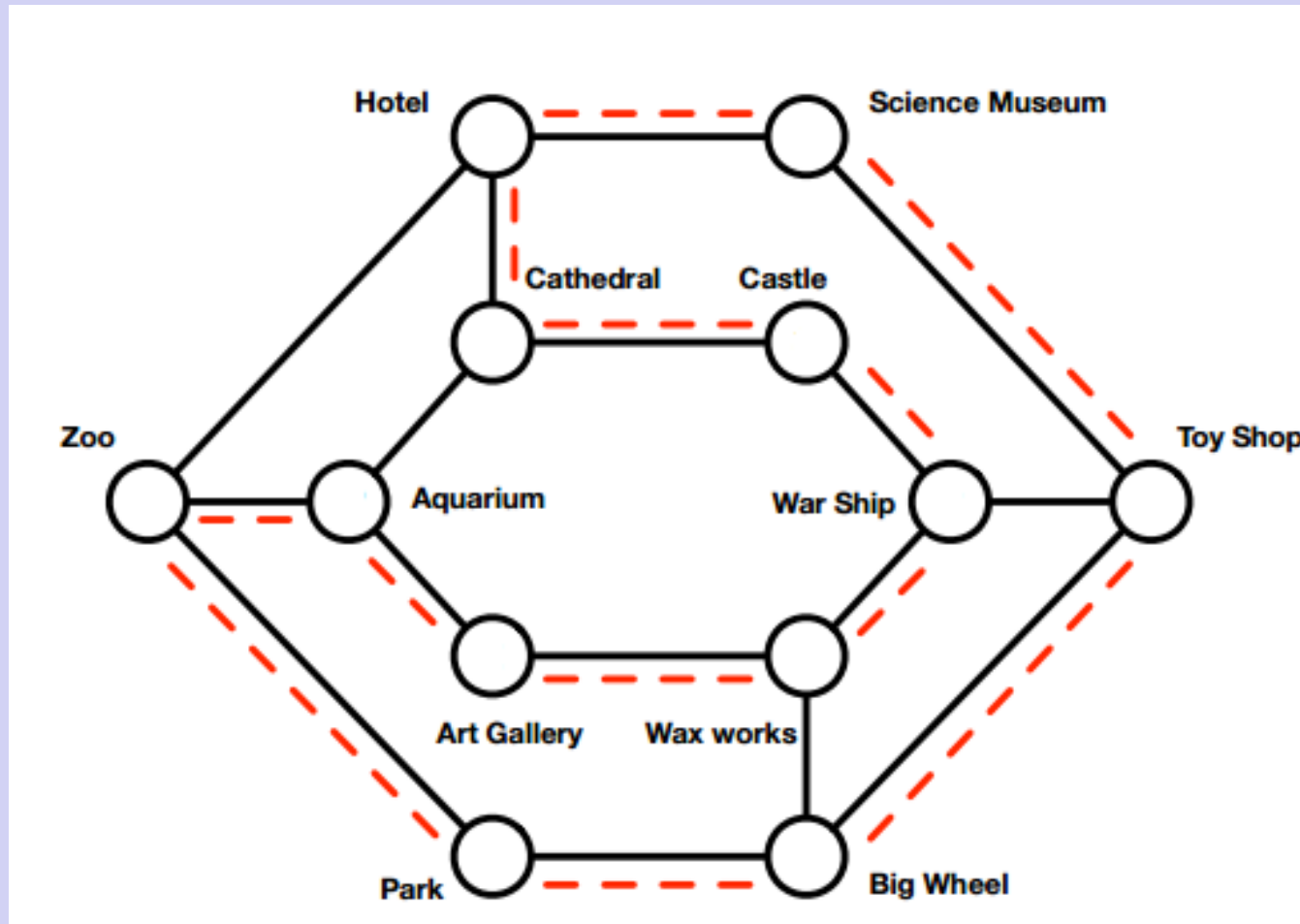
Uitdaging: Tour guide

- Oplossing?
- Aanpak?
- Moeilijkheidsgraad?

Tour guide: waar het om gaat

- Algoritmiek: opschrijven oplossing voor toekomstig bezoek (geluk gegarandeerd)
- Evaluatie: voldoet aan eisen?
- Abstractie & representatie: kaart als model (graaf), alleen relevante info
- *Verdieping: Hamiltonian cycle*

Tour Guide: een oplossing



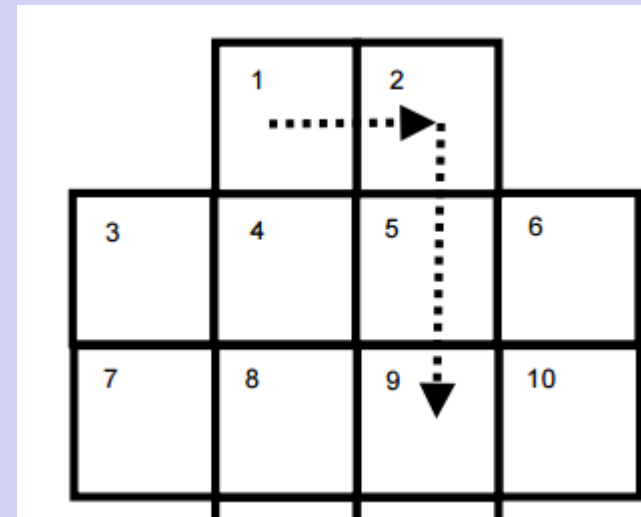
Uitdaging: Paardensprong

Zet een stuk in vakje 1

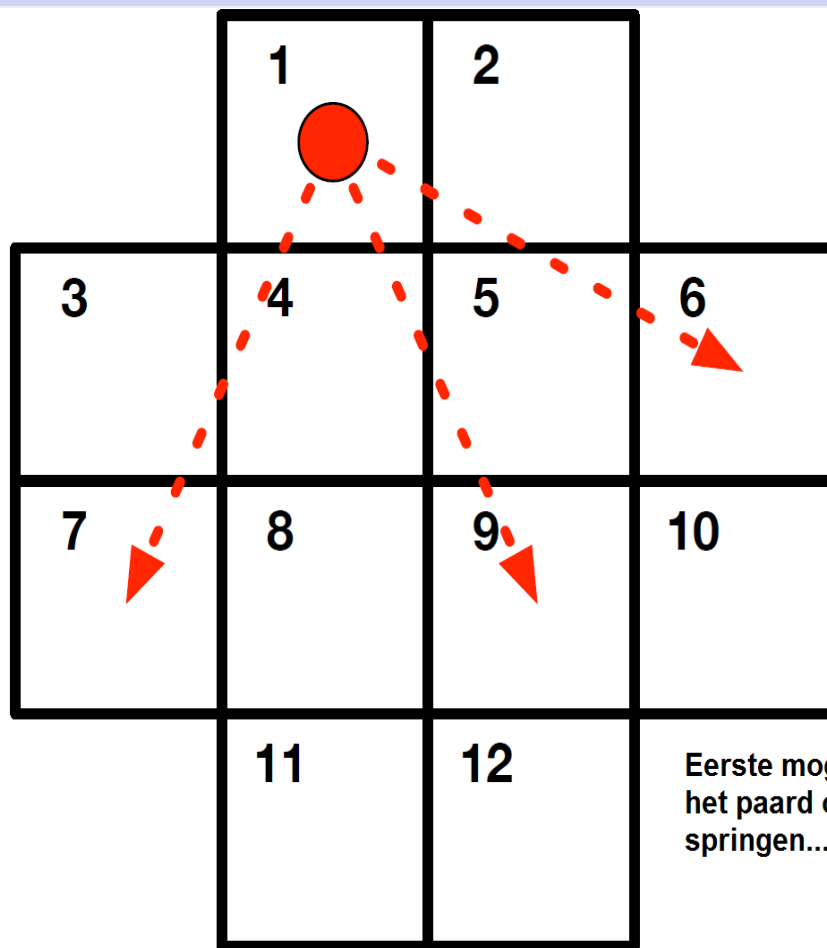
Vind een manier voor een paard om:

- elk vierkant te bezoeken
- precies één keer.

5' in tweetallen



Paardensprong

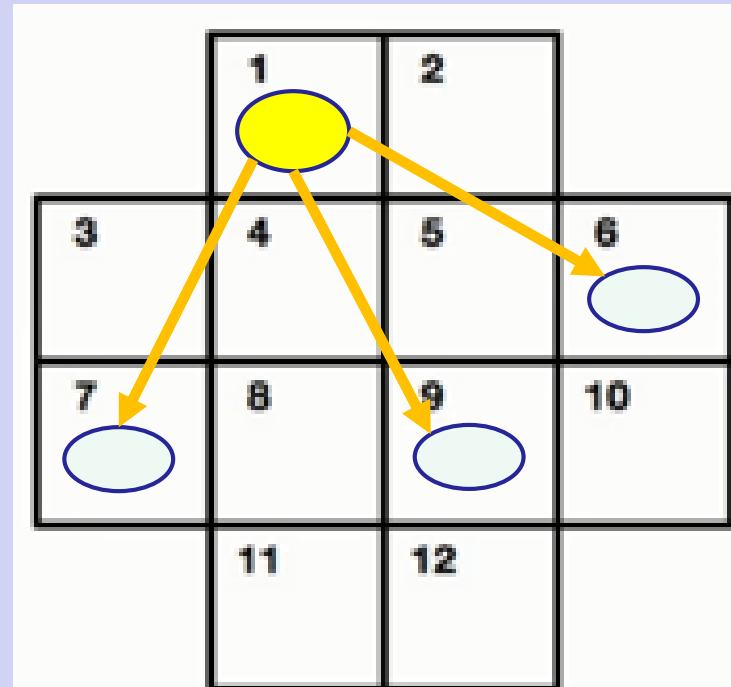
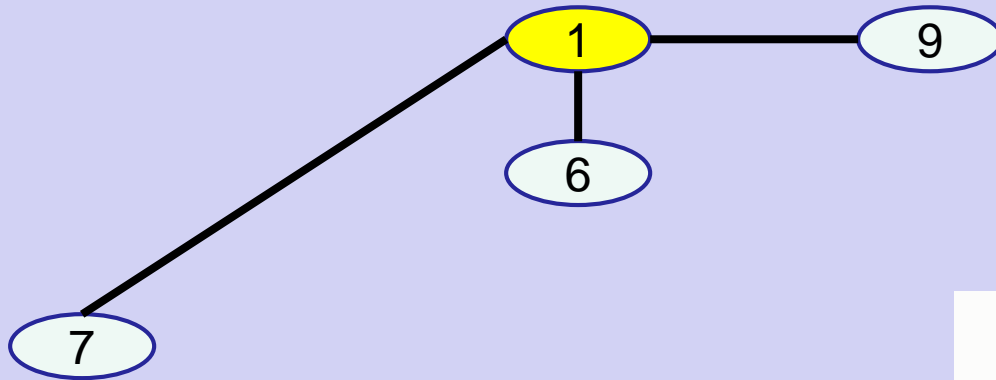


Eerste mogelijkheden voor het paard om naar toe te springen.....

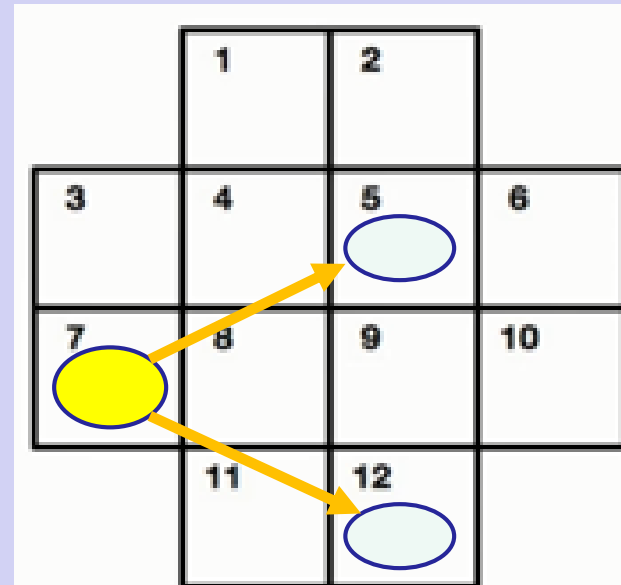
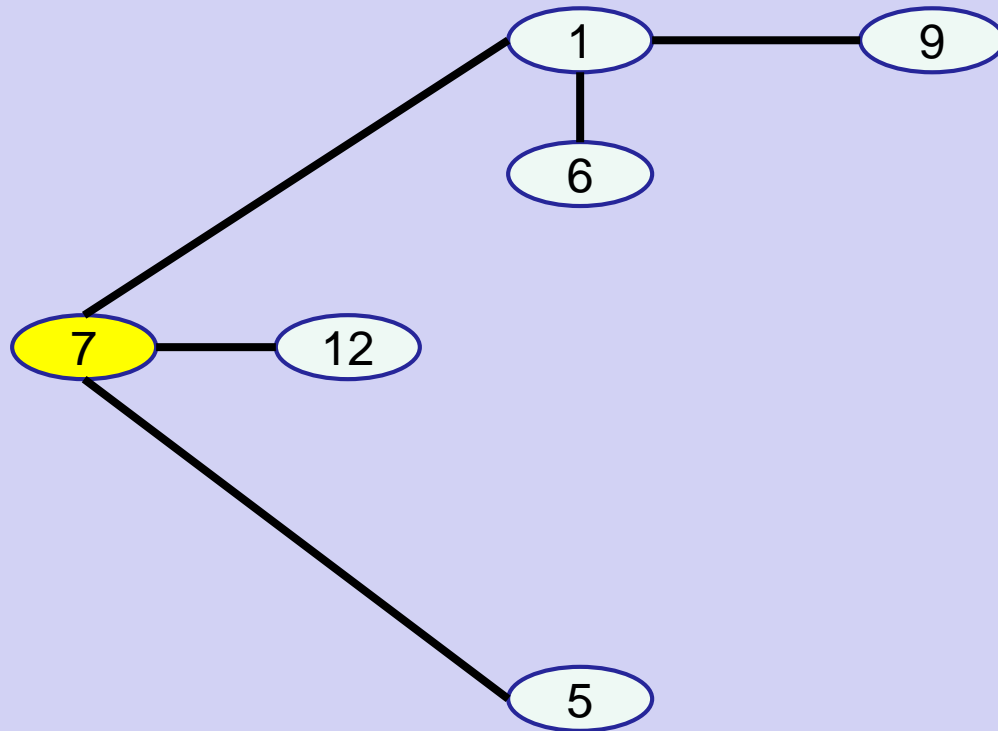
Uitdaging: Paardensprong

- Oplossingen?
- Aanpak?
- Moeilijkheidsgraad?

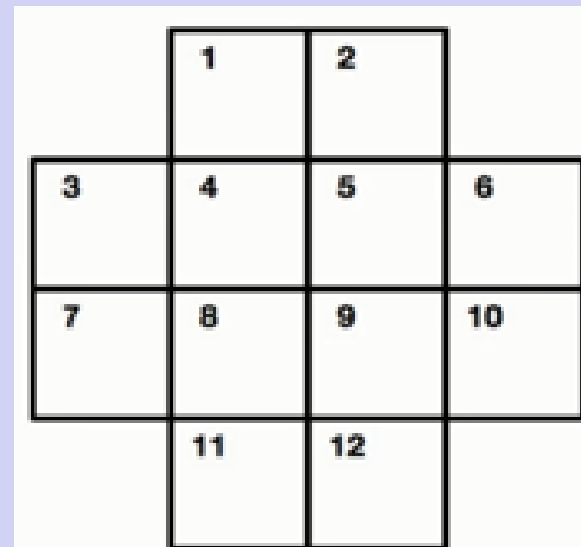
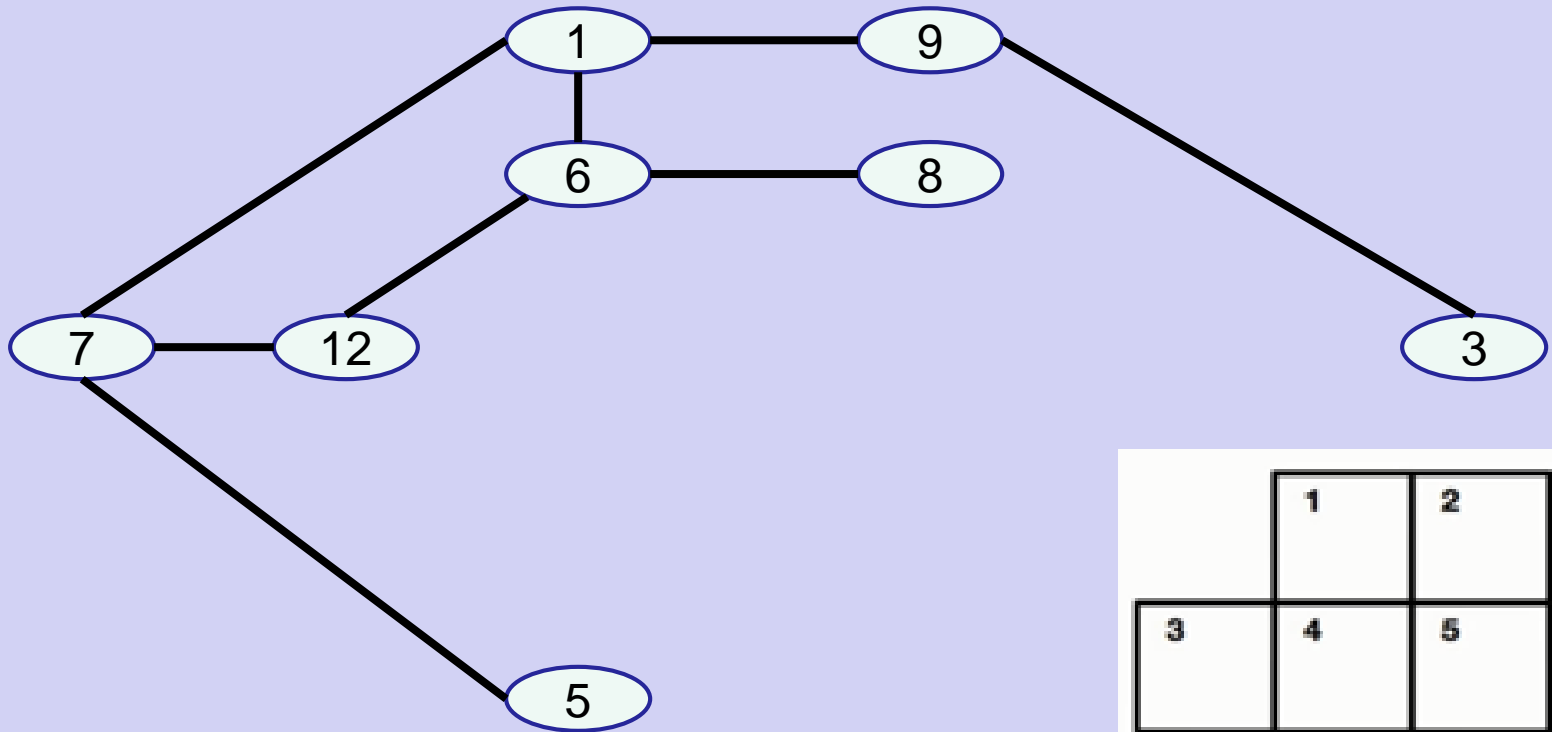
Ruiter => Graaf



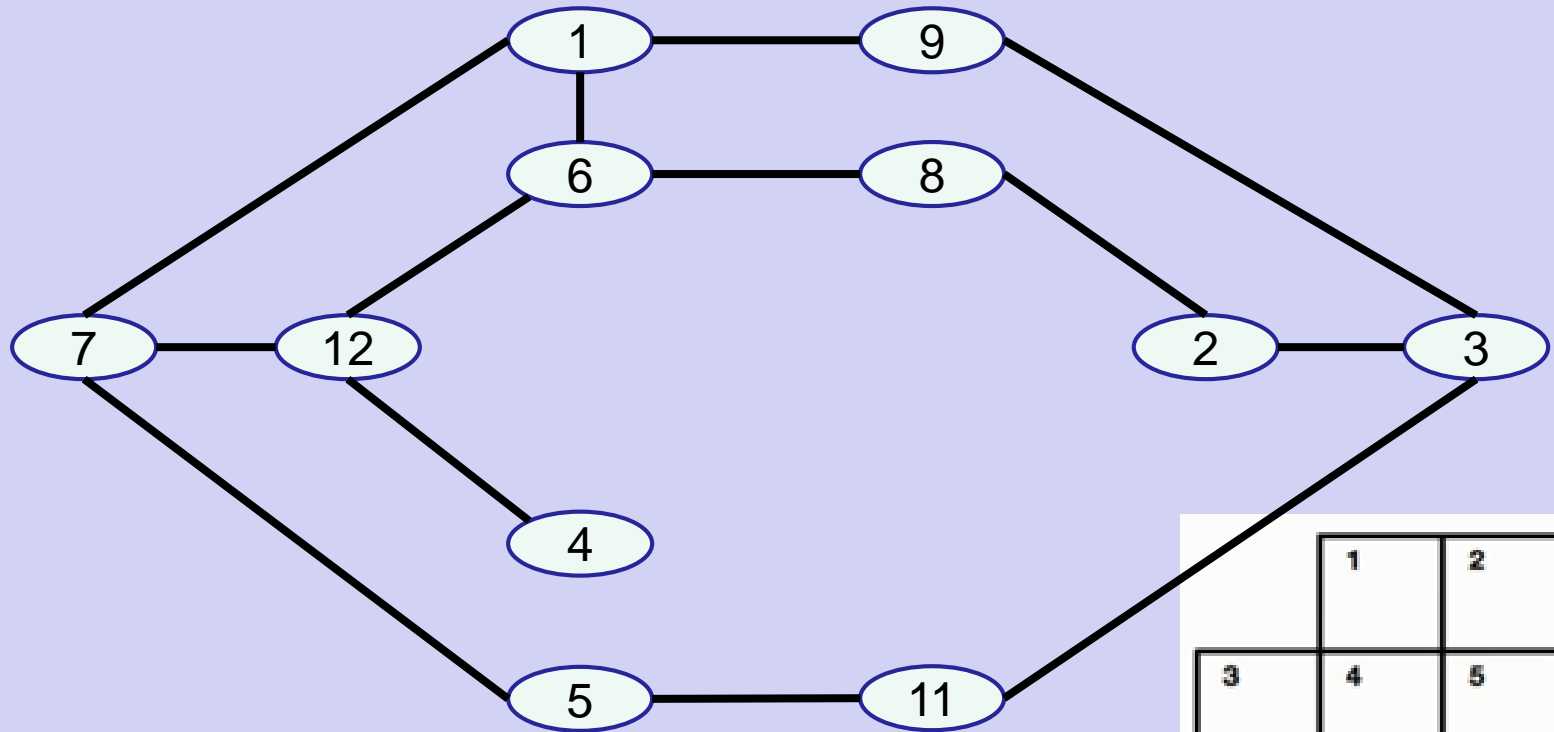
Ruiter => Graaf



Ruiter => Graaf

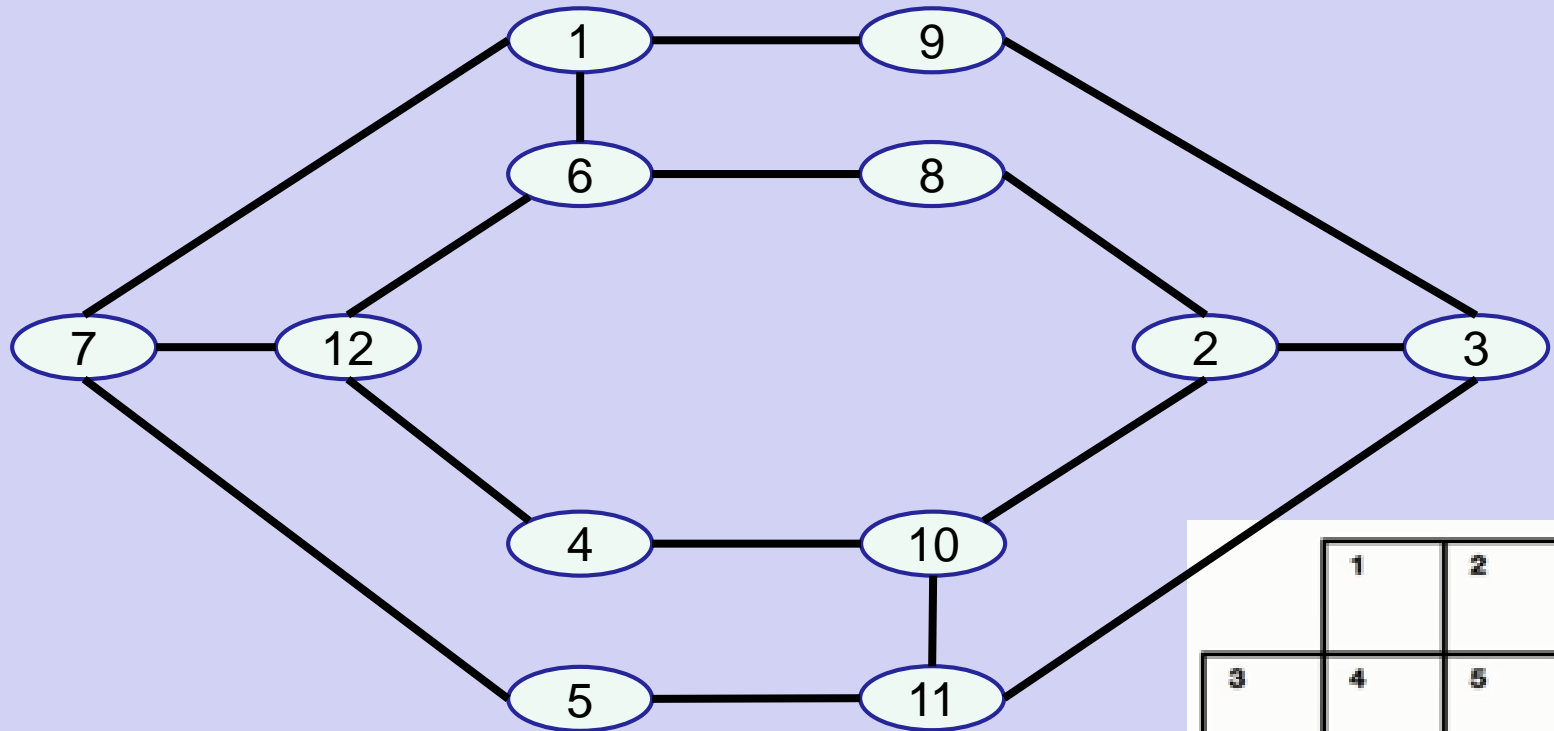


Ruiter => Graaf



	1	2	
3	4	5	6
7	8	9	10
	11	12	

Ruiter => Graaf

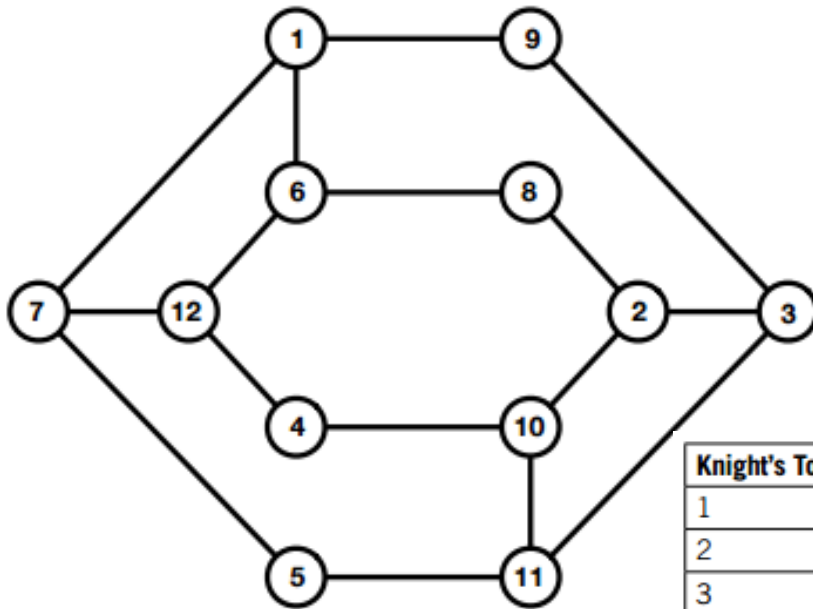


	1	2	
3	4	5	6
7	8	9	10
	11	12	

rondleiding <-> paardensprong

Paardensprong probleem	Rondleiding touristen attracties
1	Hotel
2	War ship
3	Toy Shop
4	Art Gallery
5	Park
6	Cathedral
7	Zoo
8	Castle
9	Science Museum
10	Wax Works
11	Big Wheel
12	Aquarium

Paardesprong: oplossing



1-9-3-11-5-7-12-4-10-2-8-6-1

Knight's Tour Square	Tour Guide Attraction
1	Hotel
2	War ship
3	Toy Shop
4	Art Gallery
5	Park
6	Cathedral
7	Zoo
8	Castle
9	Science Museum
10	Wax Works
11	Big Wheel
12	Aquarium

Paardensprong : waar het om gaat

Computational Thinking 'truc':

- Abstractie: details weglaten en handige representatie
- Representatie: graaf om probleem te vertegenwoordigen
- Generalisatie: bekende oplossingen gebruiken voor vergelijkbare problemen (transfer)

Paardensprong : waar het om gaat

- Niet alleen algoritmisch denken!
- De kracht abstractie!
- Keuze van representatie om probleem te vergemakkelijken
- Herkennen van problemen en oplossingsrichtingen en aan elkaar koppelen

Relevantie?

- Navigatiesystemen: steden en (1-richting verkeer) straten
- Indexeren webpagina's: sites zijn knopen (nodes), links zijn kanten (edges)
- User-interfaces: horloge of iPhone
- *Verdieping: depth-first / breadth first zoek algoritmes en (representatieve) efficiëntie*

Doelstelling materiaal

- Kennismaken met de *kunst* van het oplossen van computationele problemen
- Leuk
 - Puzzelen
 - Creatief
 - Zoeken naar oplossingen
- Serieus
 - Creëren van oplossingen



Lesmateriaal

- Focus: algoritmisch denken
- Itereren per subprobleem:
 - think-flow-code-evaluate
- Uitdagingen worden steeds complexer
- Ontwikkelen mbv flowcharts
 - Doel: Hoog niveau beschrijving
 - Opdelen in kleinere problemen

Welk ander material?

- CS unplugged

www.csunplugged.org

- Teaching London Computing

www.teachinglondoncomputing.org

- Computer Science For Fun

www.cs4fn.org

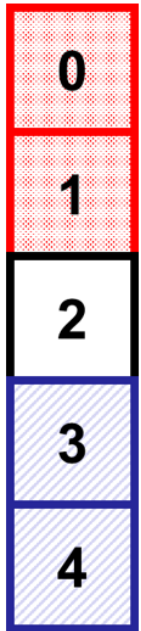
- Bever wedstrijd / Bebras

www.beverwedstrijd.nl / www.bebas.org

- Informatica olympiade / Int. Olympiad Informatics

www.informaticaolympiade.nl / www.ioinformatics.org

Unplugged



- Voordelen:
 - Leuk & uitdagend
 - Activatie leerling
 - Laag drempelig, iedereen kan begin maken
 - Samen leren (van elkaar, verschillende aanpakken/oplossingen), communiceren
- Nadelen
 - Voorbereiding docent
 - Mogelijke misconcepties / versimpeling

Alg Denk-Greenfoot lessen bestaan uit 2 delen

1. In de les: activiteit in de hoofd van de leerlingen
 - *Nadenken* zonder pc
 - Growth mindset
2. Opdrachten: oplossingen vertalen zodat computer er wat mee kan
 - Niet per se gerelateerd aan unplugged
 - Elk programmeertaal zou volstaan

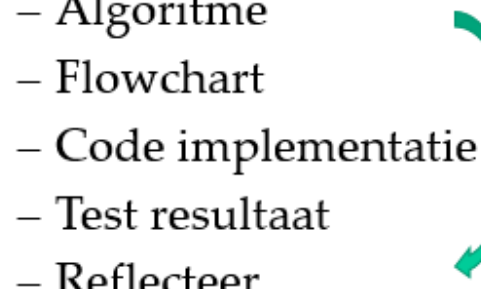
Aan de slag met opdrachten

- Kies een module (afhankelijk van jouw voorkennis) en ga daarmee aan de slag alsof je een leerling bent:
 - Stel vragen!
 - Laat zien wat je doet!
 - Laat horen wat je denkt (en vindt)!
- Je mag ook advies krijgen over andere modules

Verwachtingen voldaan?

- Unplugged didactiek
 - Samen leren / puzzelen / werken
- Greenfoot opdrachten met 'Bever'achtige puzzels
- Stappenplan

Per deelprobleem:

- 
- Algoritme
 - Flowchart
 - Code implementatie
 - Test resultaat
 - Reflecteer

Inf4All

Nascholing Alg Denken x SwissEduc - Computer S... x Beta4all x

www.beta4all.nl/prog_inf4all.htm

> Home
> Programmainformatie
> Informatica (inf4all)
> Natuurkunde (natk4all)>>
 > Wiskunde-ondersteuning
> Scheikunde (chem4all)
> Onderwijsvorm en locatie
> Toetsing
> Inschrijven en kosten
> Contact

Programma Inf4all

Het aanbod voor informatica bestaat uit acht cursussen van ieder 6 ec: twee cursussen op academisch basisoniveau voor kandidaten met een hbo-vooropleiding, en zes cursussen op gevorderd niveau.

	Semester 1	Semester 2
Academisch basisoniveau		
Grondslagen	x	
Algoritmiek		x
Gevorderd algemeen		
Advanced Object Oriented Programming	x	
Netwerken	x	
Databases en Information Retrieval		x
Gevorderd specialisatieonderwerpen		
Media, games, user experience	x	
Kunstmatige Intelligentie		x
Internet of things		x

Jaarrooster
Alle cursusbijeenkomsten vinden plaats op vrijdagen. [Download jaarrooster 2016-2017](#)
De bijeenkomsten in het eerste semester zijn gepland van 11.00 tot 16.00 uur (inclusief middagpauze), beginnend in week 36 en doorlopend t/m week 5. Het tweede semester begint in week 7 en loopt door t/m week 27, cursustijden worden in de loop van september bekend.
Vakantie: de periode tussen kerst en oud & nieuw en de goede vrijdag zijn onderwijsvrij. In de overige vakanties gaan de bijeenkomsten door.

Cursusbeschrijvingen

Cursisten over natk4all

"Heerlijke colleges met goede enthousiaste docenten!"
"Top. Inhoudelijk goed en prettig overgebracht."
"Grote kracht is de goede docenten die weten waar ze het over hebben, geweldig!"
"Leuk aan de colleges en de stof vond ik de toepassingen en het filosofische karakter."

> Meteen inschrijven?

moodle

> Log in op de ELO

allkara-en (1).jar
0,9/4,9 MB, 10 min. resterend

22:13
6-12-2016

http://www.beta4all.nl/prog_inf4all.htm

Afsluiting / Evaluatie

Vragen? Opmerkingen?

