
Lecture 2

Antwoord 8

Een expert systeem representeert het specifieke begrip van een expert binnen een systeem. Deze bestaat uit een kennis database en een inferentie component. Kennis bestaat uit feiten over de wereld en relaties tussen concepten. Inferentie vindt plaats door het manipuleren van symbolen met gebruik van bijvoorbeeld predicate logica. Verder is er data nodig. Een expert systeem is flexibel, omdat binnen bepaalde categorieën, zoals meubels in elkaar zetten, kan hetzelfde systeem gebruikt worden. Alleen het kennis component hoeft dan aangepast te worden van bijvoorbeeld tafel naar stoel.

Antwoord 9

Forward chaining houdt in, dat er met sequenties gewerkt wordt vanuit statements om tot een conclusie te komen. Het algoritme dat hierbij hoort, ziet er uit als:

1. Totdat geen enkele regel meer een nieuwe assertie produceert of totdat het concept geïdentificeerd is, geldt er voor iedere regel:
 - a) Antecedenten moeten gesteund worden door fact matching.
 - b) En als alle antecedenten gesteund worden, is de consequentie een assertie, tenzij er al een identieke assertie is.

Backward chaining houdt in, dat er gestart wordt met een hypothese en met als-dan regels wordt er terug gewerkt. Het inferentie component begint dus achteraan. Zo worden asserties gesteund. Het algoritme ziet er uit als:

2. Totdat alle hypothesen geprobeerd zijn of totdat het concept geïdentificeerd is:
 - a) Voor iedere hypothese
 - b) Voor iedere regel waarvan de consequent matcht met de hypothese
 - a) Antecedenten moeten gesteund worden door fact matching of backward chaining van een andere regel.
 - b) Als alle antecedenten gesteund worden, wordt de hypothese als waar geconcludeerd.

Antwoord 10

Je maakt gebruik van forward chaining, als je alles mogelijk wil weten van een set van feiten. Hier wil je een concept identificeren, waar je weinig informatie over hebt. De vraag is dan wat je kan deduceren vanuit observaties. Je maakt gebruik van forward chaining, als nog niet alle feiten bekend zijn en hier achter moet komen. Dan is er nog geen hypothese en irrelevante vragen worden overgeslagen.

Antwoord 11

Volgens regel 1 is het een zoogdier. Regel 5 en 6 zijn ook zoogdieren maar over die info weet je niks dus ga je door naar regel 8. Deze komt overeen met de informatie die je tot nu toe hebt. Het is dus een hoefdier. Vervolgens kijk je naar 11, daar kun je ook niks over zeggen dus kijk je verder naar 12. Deze is in overeenstemming met alle info dus is het een zebra.

Antwoord 12

Uit regel 13 blijkt dat als iets een struisvogel is, dat het dan zwart-wit is, een lange nek heeft, lange poten, kan niet vliegen en het is een vogel. Je kijkt van onder naar boven dus ga je nu naar regel 4, maar die komt niet overeen met eerdere info dus ga je naar regel 3. De struisvogel heeft dus ook veren.

Antwoord 13

Backtracking werkt door middel van herhaaldelijk genereren van mogelijke oplossingen. Als er bepaald wordt dat een mogelijke oplossing niet tot een complete oplossing kan leiden, wordt deze route verlaten. Dan wordt er terug gegaan naar het vorige keuzepunt om een nieuwe mogelijke oplossing uit te proberen. Het algoritme ziet eruit als:

1. Tot de matrix compleet is, geldt er voor iedere lege cel:
 - Voor (getal 1 tot en met 9):
 - Als er een valide mogelijke oplossing is, wordt het getal ingevuld en doorgedaan naar de volgende cel.
 - Anders wordt er een volgende mogelijke oplossing geprobeerd.

Antwoord 14

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Dit wordt gebruikt bij het hokje wat ingevuld moet worden. In het eerste lege hokje, wordt er als eerst een 1 geprobeerd. Dit kan, dus dan wordt er verder gegaan naar het middelste hokje, zonder gebruik te maken van backtracking. In het middelste hokje past een 1 niet, omdat er al een 1 in de sub matrix staat en omdat er al een 1 in de rij staat. Een 2 past hier wel en wederom kan er verder worden gegaan zonder gebruik te maken van backtracking. In het hokje daarnaast wordt een 4 ingevuld, omdat 1, 2 en 3 al in de sub matrix zitten. In het laatste open hokje kan dan alleen nog een 7. Pas in het tweede sub matrix wordt er gebruik gemaakt van backtracking, omdat er weer van voor af aan begonnen wordt.

Antwoord 15

Beide zijn basic zoek methodes, maar bij diepte eerst wordt er herhaaldelijk gebruik gemaakt het kiezen van kinderen van elke node die gekozen is. Van hieruit wordt forward gewerkt. Alternatieven worden genegeerd, zolang de huidige keuze een kans op het bereiken van het doel biedt. Als er een doodlopend pad gekozen is, wordt er terug gegaan naar het vorige keuzepunt. Dit wordt gebruikt wanneer er zekerheid is dat doodlopende paden of het doel gevonden wordt na een beperkt aantal stappen.

Bij breedte eerst worden alle paden van een niveau nagegaan voordat er naar het volgende niveau wordt gegaan. Dit is een horizontale manier van werken. Het wordt gebruikt als je werkt met diepe stambomen, maar niet wanneer alle paden naar het doel zullen leiden op dezelfde diepte.

Antwoord 16

Depth-first: S-A-B-C-B-E-D-E-F-G

Breadth-first: S-A-D-B-D-A-E-C-E-E-B-B-F-D-F-B-F-D-E-A-C-G

Antwoord 17

Een heuristisch geïnformeerd algoritme gebruikt meer informatie, zoals keuzeregels. Een voorbeeld hierbij is de afstand tussen elke node en het doel. Een naïef geïnformeerd algoritme maakt hier geen gebruik van.

Antwoord 18

Uit diepte eerst komt hill climbing en uit breedte eerst komt een beam search. Hill climbing baseert zijn keuzes op heuristieken, zoals de afstand tussen nodes het doel. Een beam search houdt in, dat er naar beneden wordt gezocht met alleen de beste w-nodes.

Antwoord 19

Dit algoritme werkt met afstanden van nodes tot het doel. Als er een mogelijke oplossing gevonden wordt, worden de rest van de paden met afstanden die groter of gelijk zijn afgekeurd. Hierdoor ligt de focus op de grootste route.

Antwoord 20

Er wordt binnen het A* algoritme gebruik gemaakt van branch-and-bound, afstanden schatten tussen nodes en het doel en redundant path removal. Met branch-and-bound wordt er gefocust op de kortste route naar het doel, door gebruik te maken van geschatte afstanden en routes die langer duren of dezelfde lengte hebben, zijn redundant. Branch-and-bound geeft informatie over hoeveel afstand er nog is tussen nodes en het doel. Afstanden zijn te schatten met verzamelde afstand geschatte afstand.