**Rīgas Tehniskā universitāte**

**Datorzinātnes un informācijas tehnoloģijas fakultāte**

**Lietišķo datorsistēmu institūts**



**studiju darbs**

mācību priekšmetā „Mākslīgā intelekta pamati”

**Izstrādāja:** Sergejs Terentjevs

IIIDBD - 4

061RDB140

**Pārbaudīja:** lekt. A.Anohina-Naumeca

2009./2010.m.g.

**Satura rādītājs**

Uzdevumu nostādne 3

1.uzdevums 4

2.uzdevums 8

3.uzdevums 9

4.uzdevums 11

5. un 6.uzdevums 15

7.uzdevums 18

8. un 9.uzdevums 21

10.uzdevums 24

secinājumi 27

bibliogrāfijas saraksts 28

Uzdevumu nostādne

1. uzdevums.

Studentam ir jāizvēlas praktiska problēma, kurā viņš jūtas eksperts. Ir jāapraksta izvēlētā problēma, jākonstruē problēmas stāvokļu telpa, ņemot vērā atbilstošā varianta ierobežojumus, jāaprēķina izveidotās stāvokļu telpas vidējais zarošanās koeficients.

2. uzdevums.

Jāaprēķina dažādu stāvokļu telpu sarežģītība, izmantojot atbilstošajā variantā dotās B un L vērtības, kā arī obligāti ir jāizmanto tās B un L vērtības, kas iegūtas pirmajā uzdevumā konstruētajai stāvokļu telpai. Tādējādi, aprēķins ir jāveic trīs dažādām vidējā zarošanās koeficienta B vērtībām, trīs reizes mainot līmeņu skaitu L. Iegūtie rezultāti ir jāatspoguļo tabulā, kā arī ir jāattēlo visas trīs līknes vienā grafikā, izmantojot logaritmisku skalu par pamatu grafiku konstruēšanai. Ņemot vērā iegūtos rezultātus un izveidotās līknes, ir jādod secinājumi par to, kā mainās ģenerēto stāvokļu skaits.

3. uzdevums.

Pirmajā uzdevumā konstruētajai stāvokļu telpai, lietojot sarakstus OPEN un CLOSED, ir jārealizē variantā norādītais neinformētas pārmeklēšanas algoritms, fiksējot atrisinājuma atrašanai nepieciešamo iterāciju skaitu. Pārmeklēšana ir jārealizē, izmantojot varianta pirmajā uzdevumā norādītās pārmeklēšanas sākuma un mērķa virsotnes.

4. uzdevums.

Pirmajā uzdevumā konstruētajai stāvokļu telpai, jākonstruē heiristiskā novērtējuma funkcija. Jāpaskaidro lieluma h(n) semantiskā jēga, vērtību skala un skaitliskās vērtības izvēle katrai virsotnei, kā arī iegūtās heiristiskā novērtējuma funkcijas vērtības ir jāatspoguļo uz attēla, kas atbilst pirmajā uzdevumā konstruētajai stāvokļu telpai.

5. un 6. uzdevumi.

Pirmajā uzdevumā konstruētajai stāvokļu telpai jārealizē divi variantā norādītie heiristiski informētas pārmeklēšanas algoritmi, lietojot sarakstus OPEN un CLOSED. Jāfiksē nepieciešamo iterāciju skaits katram no algoritmiem, un pārmeklēšanas rezultāti jāsalīdzina ar trešajā uzdevumā iegūtajiem rezultātiem (mērķa atrašanai nepieciešamo iterāciju skaitu). Pārmeklēšana ir jārealizē, izmantojot varianta pirmajā uzdevumā norādītās pārmeklēšanas sākuma un mērķa virsotnes.

7. uzdevums.

Studentam ir jāizvēlas kāda divpersonu spēle ar pilnu informāciju. Ir jādod spēles noteikumu apraksts un tās attēlojums spēles koka veidā, kā arī jānorāda uzvaru nesošie ceļi. Spēles modelēšanai ir jāizmanto vai nu Minimaksa, vai Alfa-beta algoritms.

*Piezīme:* Netiks ieskaitītas divpersonu spēles, kas tika apskatītas lekcijās, ir līdzīgas lekcijās apskatītajām spēlēm vai ir pieejamas studentiem dotajos materiālos.

8. uzdevums.

Studentam ir jāizvēlas problēmsfēra, kuras zināšanu attēlošanai būtu piemērota variantā norādītā zināšanu atspoguļošanas shēma un dotie ierobežojumi. Jāapraksta, kas tiks atspoguļots izvēlētajā problēmsfērā.

9. uzdevums.

Astotajā uzdevumā aprakstītā problēmsfēra ir jāattēlo, izmantojot variantā norādīto zināšanu atspoguļošanas shēmu. Jādod shēmas piemērotības pamatojums izvēlētajai problēmsfērai.

10. uzdevums.

Devītajā uzdevumā iegūtā zināšanu atspoguļošanas shēma jāpārraksta produkciju likumu formā. Lietojot iegūto zināšanu bāzi, jārealizē variantā norādītais izvedums, patstāvīgi izvēloties izveduma realizācijai nepieciešamus sākotnējus datus un mērķi. Vienai iterācijai jāapraksta, kā strādā parauga virzīta vadības stratēģija un unifikācijas algoritms.

*Piezīme:* Zināšanu shēmu transformācijā obligāti ir jāsaglabā devītajā uzdevumā lietotā semantika. Nedrīkst ievest jaunus objektus un saites starp tiem. Shēmas transformējot, konsekventi jāievēro viens un tas pats transformēšanas princips.

1.uzdevums

**Problēmas nosaukums:**

Portatīva datora iegāde

**Uzdevuma parametri:**

1. stāvokļu telpas līmeņu skaits = 6
2. vidējais zarošanās koeficients = 2.96
3. virsotņu skaits sākuma līmenī S = 3
4. tiešo pēcteču skaits sākuma līmeņa virsotnēm

(atsevišķi katrai virsotnei) = S1 = 3, S2 = 2, S3 = 3

1. virsotņu skaits beigu līmenī G = 5
2. tiešo priekšteču skaits beigu līmeņa virsotnēm

(atsevišķi katrai virsotnei) = G1 = 2, G2 = 3, G3= 3, G4 = 2, G5 = 3

**Problēmas apraksts:**

Mūsdienas portatīvie datori ir viena no pieprasītākajām precēm. Portatīva datora iegādei pircējiem galvenokārt ir jāvadās pēc preces iegādāšanas nolūka un naudas līdzekļiem, saskaņa ar kuriem ir jāatlasa vēlamo preci vadoties pēc produkta definējošiem raksturlielumiem: operatīvas atmiņas apjoms, cieta diska kapacitāte, ražotājs u.tml. Dota uzdevuma ir konstruēts stāvokļu telpas grafs, kurš ļauj lietotājam izvēlēties vienu vai vairākus nepieciešamos portatīvos datorus. **Stāvokļu telpa:**

**1) Stāvokļu telpas grafa attēls:**

**2) Stāvokļu telpas apraksts:**

|  |  |
| --- | --- |
| S. līmenis – Portatīva datora iegādāšanas nolūks | |
| S1 | Darba vajadzībām |
| S2 | Spēlēm |
| S3 | Lietošanai mājas apstākļos |
| A. līmenis – Video kartes atmiņas apjoms | |
| A1 | Līdz 256 MB |
| A2 | 256 – 512 MB |
| A3 | Virs 512 MB |
| B. līmenis – Operatīvas atmiņas apjoms | |
| B1 | 512 MB |
| B2 | 1 GB |
| B3 | 2 GB |
| B4 | 4 GB |
| B5 | 8 GB |
| C. līmenis – Cieta diska kapacitāte | |
| C1 | 160 GB |
| C2 | 250 GB |
| C3 | 320 GB |
| C4 | 500 GB |
| D. līmenis – Pieejama naudas summa | |
| D1 | Līdz 500 Ls |
| D2 | 500 – 1000 Ls |
| D3 | Virs 1000 Ls |
| G. līmenis – Ražotāja nosaukums | |
| G1 | Apple |
| G2 | HP |
| G3 | MSI |
| G4 | Sony |
| G5 | DELL |

**3) Vidējā zarošanās koeficienta aprēķins:**

ZF – virsotnes zarošanās faktora noteicoša funkcija.

S. līmenis = ZF(S1) + ZF(S2) + ZF(S3) = 3 + 2 + 3 = 8

A. līmenis = ZF(A1) + ZF(A2) + ZF(A3) = 5 + 5 + 5 = 15

B. līmenis = ZF(B1) + ZF(B2) + ZF(B3) + ZF(B4) + ZF(B5) = 4 + 4 + 4 + 4 + 4 = 20

C. līmenis = ZF(C1) + ZF(C2) + ZF(C3) + ZF(C4) = 3 + 3 + 3 + 3 = 12

D. līmenis = ZF(D1) + ZF(D2) + ZF(D3) = 3 + 5 + 5 = 13

G. līmenis = ZF(G1) + ZF(G2) + ZF(G3) + ZF(G4) + ZF(G5) = 0 + 0 + 0 + 0 + 0 = 0

Virsotņu skaits = 23

Vidējais zarošanas koeficients = (8 + 15 + 20 + 12 + 13 + 0) / 23 = 68 / 23 = 2.96

2.uzdevums

**Aprēķinu tabula:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **L1 =** 16 | **L2 =** 19 | **L3 =** 6 |
| **B1 =** 1,9 | 60891.0762 | 417664.2606 | 97.207971 |
| **B2 =** 16 | 1.96765E+19 | 8.05951E+22 | 17895696 |
| **B3 =** 2.96 | 52444868.9 | 1360122889 | 1014.237633 |

**Iegūto rezultātu atspoguļojums:**

Secinājumi:

Pēc grafika un aprēķinu tabulas rezultātiem var secināt, ka stāvokļu telpas sarežģītība strauji pieaug līdz ar vidēja zarošanas koeficienta palielināšanos (palielinoties loku skaitām, palielinās arī iespējamo atrisinājuma ceļu skaits), savukārt, līmeņu skaita palielināšanas rezultāta stāvokļu telpas sarežģītība pieaug proporcionāli, neliela mērā.

3.uzdevums

**Pārmeklēšanas virziens:**

no datiem virzīta

**Pārmeklēšanas algoritms:**

pārmeklēšana plašuma

**Uzdevuma parametri:**

1. pārmeklēšanas sākuma virsotne = S1
2. pārmeklēšanas mērķis =G4

**Pārmeklēšanas darbības realizācija:**

|  |  |  |
| --- | --- | --- |
| **Iterācija** | **OPEN** | **CLOSED** |
| 0. | S1 |  |
| 1. | A1 A2 A3 | S1 |
| 2. | A2 A3 B1 B2 B3 B4 B5 | S1 A1 |
| 3. | A3 B1 B2 B3 B4 B5 | S1 A1 A2 |
| 4. | B1 B2 B3 B4 B5 | S1 A1 A2 A3 |
| 5. | B2 B3 B4 B5 C1 C2 C3 C4 | S1 A1 A2 A3 B1 |
| 6. | B3 B4 B5 C1 C2 C3 C4 | S1 A1 A2 A3 B1 B2 |
| 7. | B4 B5 C1 C2 C3 C4 | S1 A1 A2 A3 B1 B2 B3 |
| 8. | B5 C1 C2 C3 C4 | S1 A1 A2 A3 B1 B2 B3 B4 |
| 9. | C1 C2 C3 C4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 |
| 10. | C2 C3 C4 D1 D2 D3 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 |
| 11. | C3 C4 D1 D2 D3 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 |
| 12. | C4 D1 D2 D3 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 |
| 13. | D1 D2 D3 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 |
| 14. | D2 D3 G2 G3 G5 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 |
| 15. | D3 G2 G3 G5 G1 G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 |
| 16. | G2 G3 G5 G1 G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 |
| 17. | G3 G5 G1 G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 G2 |
| 18. | G5 G1 G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 G2 G3 |
| 19. | G1 G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 G2 G3 G5 |
| 20. | G4 | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 G2 G3 G5 G1 |
| 21. |  | S1 A1 A2 A3 B1 B2 B3 B4 B5 C1 C2 C3 C4 D1 D2 D3 G2 G3 G5 G1 G4 |

4.uzdevums

h(n) vērtējumu skala:

Stāvokļu telpas virsotņu vērtēšanai ir izvēlēta vērtējumu skala no 1 līdz 5, kur:

* 1 – vissliktākais vērtējums,
* 2 – slikts vērtējums,
* 3 – vidējs, apmierinošs vērtējums,
* 4 – labs vērtējums,
* 5 – vislabākais vērtējums.

**h(n) vērtējumu izvēles pamatojums:**

|  |  |  |
| --- | --- | --- |
| **Virsotnes apzīmējums** | **h(n) vērtējums** | **Paskaidrojums** |
| S1 | 5 | Portatīvais dators ir labi piemērots šāda mērķa īstenošanai. Tās ļauj optimizēt, organizēt un vadīt ikdienas darbu. |
| S2 | 3 | Nav īsti piemērots doto nolūku īstenošanai. Jaunāko spēļu gadījumā dators ir jāmodernizē, ko būtība nav iespējams izdarīt. |
| S3 | 4 | Var tikt pielietots arī mājas apstākļos, taču tā pilna funkcionalitāte netiek izmantota. |
| A1 | 3 | Video karte nav piemērota jaunāko spēļu atbalstām. |
| A2 | 4 | Apmierina visas spēlētāju prasības, taču atsevišķos gadījumos tas jauda var būt nepietiekoša. |
| A3 | 5 | Der visam spēlēm un var tikt pielietota darbam grafikas konstruējošam programmām kā AutoCAD, kur ir vēlama augsta pikseļu izšķirtspēja. |
| B1 | 1 | Saskaņa ar mūsdienu operētājsistēmu un programmatūras prasībām šāds atmiņas apjoms nav pietiekams. |
| B2 | 2 | Paralēli pildoties vairākiem atsevišķiem procesiem, atmiņas apjoms var tikt aizņemts, kā rezultātā dators sāks izmantot virtuālo atmiņu. |
| B3 | 3 | Atmiņas apjoms ir apmierinošs un vienkārša lietotāja prasībām pietiekams. |
| B4 | 4 | Atmiņas apjoms ir pietiekams jebkura lietotāja vajadzību apmierināšanai, izmantojot to var paralēli darbināt spēles, filmas, pārlūkprogrammu u.t.t. |
| B5 | 5 | Ļauj īstenot visas 64 bitu operētājsistēmu iespējas. |
| C1 | 2 | Cieta diska ietilpība nav pietiekama datora lietošanai mājas apstākļos. |
| C2 | 3 | Ietilpības apjoms ir apmierinošs. |
| C3 | 4 | Ietilpības apjoms ir pietiekošs ilga laika lietošanai. Izmantojot to var veidot filmu u.c. multimediju datņu kolekciju. |
| C4 | 5 | Ietilpības apjoms ir pietiekošs gan multimediju datņu, gan arī rezerves kopiju uzturēšanai. |
| D1 | 3 | Var iegādāties vidējas funkcionalitātes (kvalitātes) datorus, kuri spēj apmierināt vairākas lietotāju vajadzības. |
| D2 | 4 | Var iegadāties labas kvalitātes portatīvos datorus, kuri funkcionalitātes ziņā neatpaliek no augstas kvalitātes datoriem. |
| D3 | 5 | Var iegādāties augstas kvalitātes portatīvos datorus. |
| G1 | 5 | Kompānija piedāvā augstās kvalitātes portatīvos datorus. |
| G2 | 4 | Kompānija piedāvā portatīvos datorus dažādu vajadzību segmentu apmierināšanai, sākot ar vidējas kvalitātes un beidzot ar augstas kvalitātes datoriem. |
| G3 | 4 | Kompānija piedāvā portatīvos datorus dažādu vajadzību segmentu apmierināšanai. |
| G4 | 5 | Kompānija piedāvā augstās kvalitātes portatīvos datorus. |
| G5 | 4 | Kompānija piedāvā portatīvos datorus dažādu vajadzību segmentu apmierināšanai. |

Heiristiskā novērtējuma funkcijas aprēķins:

|  |  |  |  |
| --- | --- | --- | --- |
| **Virsotnes apzīmējums** | **h(n) vērtējums** | **g(n) vērtība** | **f(n) vērtība** |
| S1 | 5 | 0 | 5 |
| S2 | 3 | 0 | 3 |
| S3 | 4 | 0 | 4 |
| A1 | 3 | 1 | 4 |
| A2 | 4 | 1 | 5 |
| A3 | 5 | 1 | 6 |
| B1 | 1 | 2 | 3 |
| B2 | 2 | 2 | 4 |
| B3 | 3 | 2 | 5 |
| B4 | 4 | 2 | 6 |
| B5 | 5 | 2 | 7 |
| C1 | 2 | 3 | 5 |
| C2 | 3 | 3 | 6 |
| C3 | 4 | 3 | 7 |
| C4 | 5 | 3 | 8 |
| D1 | 3 | 4 | 7 |
| D2 | 4 | 4 | 8 |
| D3 | 5 | 4 | 9 |
| G1 | 5 | 5 | 10 |
| G2 | 4 | 5 | 9 |
| G3 | 4 | 5 | 9 |
| G4 | 5 | 5 | 10 |
| G5 | 4 | 5 | 9 |

**Stāvokļu telpa:**

4

3

5

6

5

4

7

6

5

4

3

8

5

6

7

9

8

7

10

10

9

9

9

5. un 6.uzdevums

**Uzdevuma parametri:**

1. pārmeklēšanas sākuma virsotne = S1
2. pārmeklēšanas mērķis =G4

**Pārmeklēšanas virziens:**

no datiem virzīta

**Pārmeklēšanas algoritms:**

vislabākā stāvokļa meklēšana

**Stara platums:**

[ierakstiet Jūsu datus]

**Pārmeklēšanas darbības realizācija:**

|  |  |  |
| --- | --- | --- |
| **Iterācija** | **OPEN** | **CLOSED** |
| 0. | S1[5] |  |
| 1. | A3[6] A2[5] A1[4] | S1[5] |
| 2. | B5[7] B4[6] A2[5] B3[5] A1[4] B2[4] B1[3] | S1[5] A3[6] |
| 3. | C4[8] C3[7] B4[6] C2[6] A2[5] B3[5] C1[5] A1[4] B2[4] B1[3] | S1[5] A3[6] B5[7] |
| 4. | D3[9] D2[8] C3[7] D1[7] B4[6] C2[6] A2[5] B3[5] C1[5] A1[4] B2[4] B1[3] | S1[5] A3[6] B5[7] C4[8] |
| 5. | G1[10] G4[10] G2[9] G3[9] G5[9] D2[8] C3[7] D1[7] B4[6] C2[6] A2[5] B3[5] C1[5] A1[4] B2[4] B1[3] | S1[5] A3[6] B5[7] C4[8] D3[9] |
| 6. | G4[10] G2[9] G3[9] G5[9] D2[8] C3[7] D1[7] B4[6] C2[6] A2[5] B3[5] C1[5] A1[4] B2[4] B1[3] | S1[5] A3[6] B5[7] C4[8] D3[9] G1[10] |

**Pārmeklēšanas virziens:**

no datiem virzīta

**Pārmeklēšanas algoritms:**

kalnā kāpšana

**Stara platums:**

[ierakstiet Jūsu datus]

**Pārmeklēšanas darbības realizācija:**

|  |  |  |
| --- | --- | --- |
| **Iterācija** | **OPEN** | **CLOSED** |
| 0. | S1[5] |  |
| 1. | A3[6] | S1[5] A1[4] A2[5] |
| 2. | B5[7] | S1[5] A1[4] A2[5] A3[6] B1[3] B2[4] B3[5] B4[6] |
| 3. | C4[8] | S1[5] A1[4] A2[5] A3[6] B1[3] B2[4] B3[5] B4[6] B5[7] C1[5] C2[6] C3[7] |
| 4. | D3[9] | S1[5] A1[4] A2[5] A3[6] B1[3] B2[4] B3[5] B4[6] B5[7] C1[5] C2[6] C3[7] C4[8] D1[7] D2[8] |

Algoritmu salīdzinājums:

„Kalnā kāpšanas” algoritms vadoties pēc virsotņu heiristiskiem vērtējumiem (dotajā uzdevuma labāka virsotne ir ar lielāko heiristisko vērtējumu) katra nākama iterācijā sarakstā OPEN ievieto tikai vienu virsotni. Šādai pārmeklēšanas stratēģijai ir gan savi trūkumi, gan arī priekšrocības. Par priekšrocību var minēt atmiņas resursu ietaupīšanu, kas ir lietderīgi lielas stāvokļu telpas gadījumā. Taču algoritms garantē mērķa atrašanu tikai ja heiristika nav maldīga. Nonākot Plato problēmas situācijā, algoritms nespēj atkāpties (saraksta ir tikai viena virsotne), beidzot darbu nesasniedzot mērķi. Dota uzdevumā algoritms ir nonācis šāda situācijā, kurā tās nespēj izvēlēties vienu no labākam virsotnēm, jo abu virsotņu heiristiski vērtējumi ir vienādi.

„Vislabākā stāvokļa meklēšanas” algoritms katrā nākama iterācija uztur neapskatītas virsotnes, kas ļauj tām realizēt atkāpšanos. Šādi dotajā uzdevuma algoritms spēj atrast mērķi, atkāpjoties no virsotnes G1 (5. un 6. iterācijā).

Ceturtā uzdevuma uzdotas stāvokļu telpas pārmeklēšanai ir efektīvāk pielietot „vislabākā stāvokļa meklēšanas” algoritmu, jo stāvokļu telpa ir neliela un algoritms garantē mērķa sasniegšanu.

7.uzdevums

**Izmantotais algoritms:**

Minimaksa algoritms

Spēles nosaukums:

Uguns bumbas

Spēles apraksts:

Spēle paredz divu uguns bumbu patvaļīgo ievietošanu laukumā 3x3. Katra gājienā spēlētājiem ir jāpārvieto uguns bumbu vai nu pa labi, vai nu pa kreisi, vai arī uz augšu vai uz leju, kas tiek izdarīts atkarība no uguns bumbas atrašanas spēles laukumā. Spēlētājs nevar pārvietot uguns bumbu pa diagonāli. Pārvietojot uguns bumbu tā atstāj nospiedumu, kurā vairs nevar atkārtoti ievietot uguns bumbu. Zaudē tās spēlētājs, kuram nav atlicis brīvo lauciņu uguns

bumbu pārvietošanai.

**Spēles koks:**



Spēles analīze:

Spriežot pēc algoritma izpildes rezultātā iegūtiem uzvaru nesošiem ceļiem ir redzams, ka ar lielo varbūtību uzvarēs tieši tās spēlētājs (maksimizētājs), kurš veic gājienu otrais un pretēji maksimizētājs ar lielo varbūtību zaudēs ja izdarīs gājienu pirmais. Izņēmuma situācija ir iespējama, jā maksimizētājs ceturtajā līmenī izdarīs kļūdainu gājienu, kā rezultātā viņa zaudēšanas iespējamība strauji pieaug – uzvaru nesošo ceļu skaits strauji samazinās (uzvaras izredzes iespēja kļūst „50 pret 50”).

8. un 9.uzdevums

**Zināšanu atspoguļošanas shēma:**

konceptuālais grāfs

Problēmas nosaukums:

Preču piegāde

Problēmas apraksts:

Uzdevumā ar konceptuālo grāfu palīdzību tiek atspoguļotas zināšanas par preču piegādes darbības jomu. Tajos tiek atspoguļots kopējs preces piegādes process, kas tiek dalīts (apskatīts) atsevišķos izpildes posmos.

**Uzdevuma parametri:**

Freimiem:

|  |  |
| --- | --- |
| **Kopējais freimu skaits** | [freimu skaits] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |
| [atsevišķa freima nosaukums] | [slotu skaits freimā] |

**Semantiskajam tīklam vai konceptuālajam grafam:**

**Konceptu skaits:** 14

**Scenārijam:**

**Darbību skaits:** [ierakstiet Jūsu scenārijā esošo darbību skaitu]

**Atspoguļojums:**



**Shēmas piemērotības pamatojums:**

Konceptuālie grāfi ļauj atspoguļot daudzveidīgo informāciju par konceptiem. Izmantojot tos ir iespējams veikt preču piegādes darbības jomas dekompozīciju, apskatot katru atsevišķo darbības izpildes posmu, kas varbūt lietderīgi, piemēram, ja tiek meklēts risinājums darba laika optimizēšanai.

10.uzdevums

**Izveduma tips:**

inversais

**Likumi:**

|  |  |  |
| --- | --- | --- |
| **Likuma numurs** | **IF daļa** | **THEN daļa** |
| 1 | Saņemt(A) | Piegādātājs(B) |
| 2 | Piegādātājs(B) | Pasūtītājs(C) |
| 3 | Pasūtītājs(C) | Adrese(D) |
| 4 | Adrese(D) | Sasniegt(E) |
| 5 | Sasniegt(E) | Piegādātājs(B) |
| 6 | Piegādātājs(B) | Klauvēt(F) |
| 7 | Klauvēt(F) | Pasūtītājs(C) |
| 8 | Pasūtītājs(C) | Atvērt(G) |
| 9 | Atvērt(G) | Piegādātājs(B) |
| 10 | Piegādātājs(B) | Atdod(H) |
| 11 | Atdod(H) | Prece(I) |
| 12 | Prece(I) | Pasūtītājs(C) |
| 13 | Pasūtītājs(C) | Nauda(J) |

**Sākotnēji dati:**

J

**Mērķis:**

D

**Izveduma realizācija:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Iterācija** | **Darba atmiņa** | **Konfliktu kopa** | **Iedarbinātais likums** |
| 0. | J | 13 | 13 |
| 1. | J, C | 13, 12, 7, 2 | 12 |
| 2. | J, C, I | 13, 12, 7, 2, 11 | 11 |
| 3. | J, C, I, H | 13, 12, 7, 2, 11, 10 | 10 |
| 4. | J, C, I, H, B | 13, 12, 7, 2, 11, 10, 9, 5, 1 | 9 |
| 5. | J, C, I, H, B, G | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8 | 8 |
| 6. | J, C, I, H, B, G | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8 | 7 |
| 7 | J, C, I, H, B, G, F | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8, 6 | 6 |
| 8 | J, C, I, H, B, G, F | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8, 6 | 5 |
| 9 | J, C, I, H, B, G, F, E | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8, 6, 4 | 4 |
| 10 | J, C, I, H, B, G, F, E, D | 13, 12, 7, 2, 11, 10, 9, 5, 1, 8, 6, 4, 3 | beigas |

**Vienas iterācijas apraksts:**

Trešās iterācijas gadījumā, pie esošiem paraugiem J, C, I tiek ievietots jauns paraugs H. Turpmāk tiek pārbaudīta parauga sakrišana ar THEN daļu, kā rezultātā tiek atklāts 10 likums. Dotais likums tiek ievietots konfliktu kopa un iedarbināts.

secinājumi

Darbs, galvenokārt, bija interesants ar to, ka ļāva praktiski apgūt mākslīga intelekta pamatus. Vadoties pēc 8., 9. un 10. uzdevumu izpildes ir radies praktisks priekšstats par zināšu atspoguļošanas viediem, kas var tikt pielietoti zināšanu bāzes izstrādei ekspertu, intelektuālam sistēmām. Konceptuālos grafu konstruēšana ļāva aizdomāties par filozofijas (siloģismu) u.c. nozaru nozīmi mākslīga intelektā. Interesanti bija pielietot algoritmus, kuri ļauj risināt pārmeklēšanas problēmas dažādos veidos, atklāt to trūkumus un priekšrocības. Minimaksa algoritma praktiska pielietošana ļāva radīt stratēģijas spēlēs konkurentu apspēlēšanai.

bibliogrāfijas saraksts

* E-apmācības sistēma RTU portālā (ortus.rtu.lv):

E-studiju kurss: Mākslīgā intelekta pamati, Dr.habil.sc.ing., prof. Jānis Grundspeņķis, Dr.sc.ing., lekt. Alla Anohina.