UI prednaska c2

Realne problem – problem najdenia cesty

Problem naplanovania najvyhodnejsej cestovnej trasy z mesta A do mesta B

* Stavy: mesta, ktore sa uvazuju pri hladani
* Zaciatocny stav: mesto A
* Operatory: mozne presuny z jedneho mesta do druheho (existuje cesta na mape)
* Cielovy test: “sme v meste B?”
* Cena cesty: aplikacia operatora, t.j. presun z jedneho mesta do druheho, ma cenu rovnajucu sa vzdialenosti medzi tymito mestami

**14-15 problem**

Vyprava o hrach a puzzles

**Hladanie riesenia**

* Je pristup k rieseniu problemov, pri ktorom nevychadzame z algoritmu riesenia problem
* Bud ho nepozname alebo pozname

Function VSEOBECNE\_HLADANIE(problem, strategia)

Returns riesenie alebo neuspech

Inicializuj strom hladania pouzitim zaciatocneho stavu z problem

^je to na dokumentovom

**Stavovy priestor a graf (strom) hladania**

* Reprezentacia uzla:
  + Zodpovedajuci stav zo stavoveho priestoru
  + Uzol v strome hladania z ktoreho sad any uzol vygeneroval
  + Operator, ktory sa aplikoval pri generovani uzla (parent uzol)
  + Pocet uzlov na ceste z korena do daneho uzla (hlbka uzla)
  + Cena cesty zo zaciatocneho uzla do daneho uzla

Datatype UZOL

Components: STAV, RODICOVSKY\_UZOL, OPERATOR, HLBKA, CENA\_CESTY

**Strategie hladania**

* **Neinformovane (slepe)**
  + Nemaju k dispozicii nejaku doplnujucu informaciu o probleme
  + Poradie generovania stavov zavisi iba od informacii ziskanych hladanim a nieje ovplyvnene ani nepreskumanou castou grafu ani vlastnostami cieloveho stavu
* **Informovane (heuristicke)**
  + Maju k dispozicii informacie o proleme

**Heruistika v informatike**

* Postup, ktory zvycajne vedie k dobremu rieseniu, avsak nezarucuje, ze sa najde najlepsie riesenie, ani ze sa najde v kratkom case, ani ze sa vobec najde

**Strategie hladania – vlastnostni**

* Uplnost – zarucuje hladanie s danou strategiou, ze sa najde riesenie, ak existuje?
* Casova zlozitost – ako dlho trva, kym sa najde riesenie?
* Pamatova zlozitost – kolko pamati treba na vykonanie hladania?
* Pripustnost (optimalnost) – najde sa pomocou danej strategie najlepsie riesenie, ak existuje aspon jedno riesenie?

**Slepé stratégie**

* Do sirky (Breadth-first search)
  + Obojsmerne
* Do hlbky (DFS)
  + Obmedzene
  + Iterativne sa prehlbujce
  + Do hlbky s navratom
* Rovnomerna cena (varianta do sirky)

**Hladanie do sirky**

* Uplne, pripustne, exponencionalna zlozitost
* Nove uzly sa pridavaju na koniec OKRAJa (zoznam nasich nerozvinutych uzlov)
* Dolezite parameter:
  + Maximalny pocet nasledovnikov ktorehokolvek stavu -> factor vetvenia b prehladavaneho stromu
  + Minimalna dlzka cesty medzi pociatocnym a cielovym stavom ->hlbka d najplytsieho cieloveho uzlaa v strome
* b: vetviaci factor
* d: hlbka najplytsieho cieloveho uzla
* Hladanie do sirky je:
  + Uplne
  + Optimalne, ak je krok 1
* Pocet vygenerovanych uzlov
  + 1+b+b^2+…+b^d = (b^(d+1)-1)/(b-1) = O(b^d)

**Hľadanie do hĺbky (Depth-first search)**