

UNIVERSITAT DE LLEIDA

Escola Politècnica Superior

Grau en Enginyeria Informàtica

INTEL·LIGÈNCIA ARTIFICIAL

PAC-MAN ALGORITHMS

Professorat: Carlos Ansotegui , Josep Pon

Alumnat: Sergi Cervera, Santi Gatiús

Curs 2015-2016

Decisions de disseny

Les decisions que s'han pres a l'hora d'implementar els diversos algoritmes, han estat les de optimitzar el codi per tal de aconseguir expandir el menor nombre de nodes i així consumir menys recursos i l'altra ha estat la de utilitzar el menor nombre de codi possible, i mitjançant les característiques de cada algoritme, que el codi s'adaptés a l'algoritme al qual fèiem referència, passant-li aquestes particularitats com a paràmetres; un exemple, seria el de passar per paràmetre una "queue" o una "stack" en funció de si el que volem executar un BFS o un DFS, a través d'un mètode més genèric; com blindSearch que és el que em utilitzat nosaltres en aquest cas.

En la implementació de l'algoritme DLS, tenint en compte que els cost de pas en el nostre problema es 1, s'ha utilitzat la variable cost, per a representar la profunditat de l'algoritme. En el cas contrari, hauríem d'haver modificat el fitxer node.py he incorporar un nou atribut a aquest objecte, el qual seria per a reflexar la profunditat.

Avaluació experimental

A continuació, es mostren el nombre de nodes expandits i el temps (expressat en segons) que s'ha trigat per trobar el camí, dels diferents tipus d'algoritmes implementats en funció dels diversos mapes en els quals s'han executat.

	tinyMaze		smallMaze		bigMaze		openMaze	
	Nodes	Temps	Nodes	Temps	Nodes	Temps	Nodes	Temps
BFS	15	0.062	90	0.104	617	0.842	679	0.250
DFS	14	0.101	59	0.122	390	0.842	315	0.842
TBFS	36	0.114	450	0.119	30387	1.162	8430	0.371
TDFS	Timeout		Timeout		Timeout		Timeout	
TIDS	398	0.109	7847658	1:15.133	Timeout		Timeout	
UCS	15	0.120	92	0.193	620	0.822	682	0.305
A*	15	0.149	91	0.242	619	0.804	682	0.274
mandH								
A*	15	0.124	91	0.218	619	1.009	682	0.321
eucdH								
BFSH	8	0.117	39	0.174	466	0.957	89	0.304
mandH								
BFSH	8	0.116	39	0.174	471	0.768	54	0.213
eucdH								

Per finalitzar, podem concloure que en aquest cas concret, el millor algoritme dels que em implementat es el BFSH, ja que és el que expandeix menys nodes i tarda menys temps en trobar la solució al problema.