

人工智能概论 (张白一老师)

实验一

孙 晖

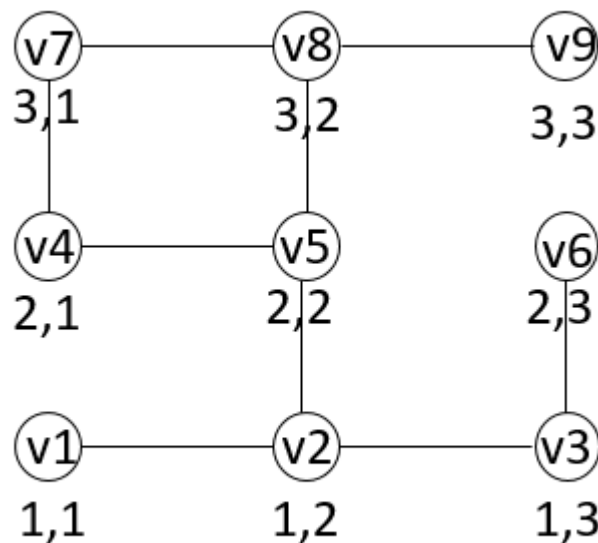
1513012 班 15130120141

实验一：

按照张白一老师的要求，学生自己规定题目，自己制定 **3*3** 迷宫，要求迷宫至少有两条路能够抵达终点，同时要写出数据结构和状态树，以及 **5** 个表（**open** 表、**closed** 表、**res** 表、**mark** 表、**fail** 表），然后附上程序，最后给出运行结果的截图。

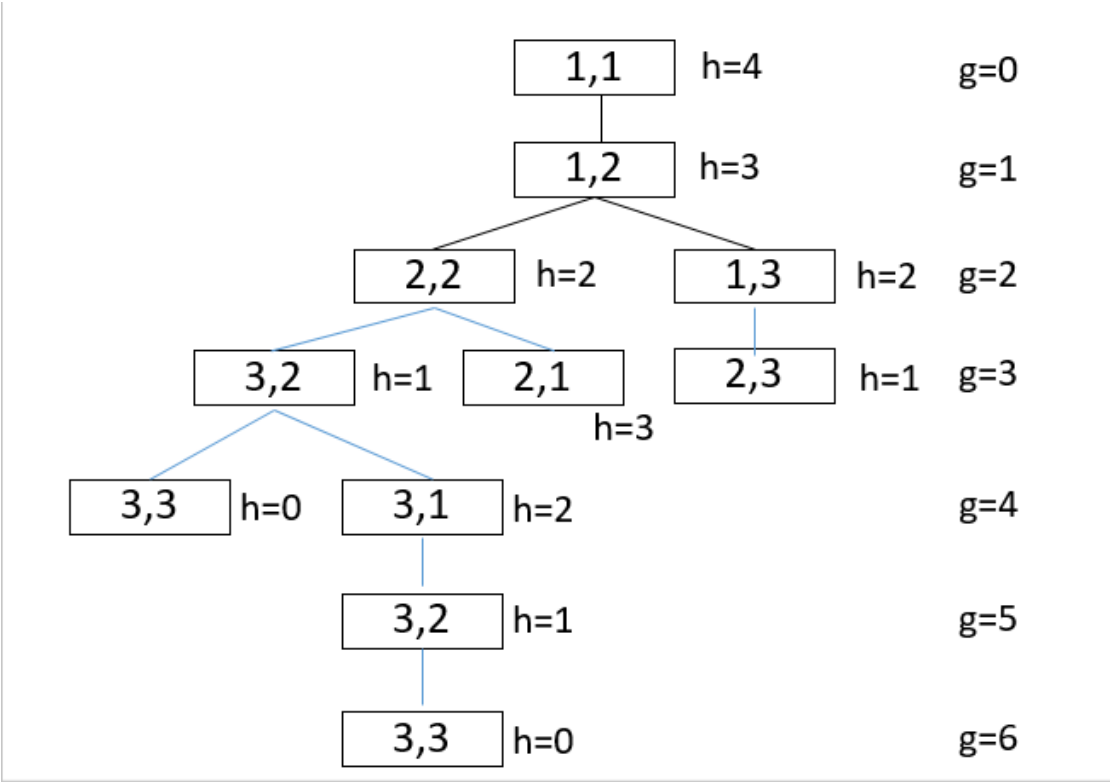
答：

如下图所示，**v1** 为迷宫的入口，**t** 为迷宫的出口，一共有两条路可以抵达出口，分别为 **v1-v2-v5-v8-t**，以及 **v1-v2-v5-v4-v7-v8-t**，其中前者为最优路径。



我规定的操作集为“上、右、下、左”。

状态空间树如下图所示：



实际上，找到最短路径后就没有再展开右边的节点了。

open 表

节点	父节点
V1	NULL
V2	V1
V5	V2
V3	V2
V8	V5
V4	V5
V9	V8

closed 表

编号	节点	父节点
7	V9	V8
6	V4	V5
5	V8	V5
4	V3	V2
3	V5	V2
2	V2	V1
1	V1	NULL

mark 表

目标节点标记
V9

res 表

路径节点
V1
V2
V5
V8
V9

fail 表

NULL

路径为：v1-v2-v5-v8-v9.

代码：

DOMAINS

state=symbol

DATABASE-mydatabase

open(state,integer)

closed(integer,state,integer)

res(state)

mark(state)

fail_

PREDICATES

solve

search(state,state)

result

searching

step4(integer,state)

step56(integer,state)

equal(state,state)

repeat

resulting(integer)

rule(state,state)

road(state,state)

GOAL

solve.

CLAUSES

solve:- search(v1,v9),result.

search(Begin,End):-

retractall(_,mydatabase),

```
assert(closed(0,Begin,0)),
assert(open(Begin,0)),
assert(mark(End)),
repeat,
searching,!.
```

result:-

```
not(fail_),
retract(closed(0,_,0)),
closed(M,_,_),
resulting(M),!.
```

result:- beep,write("I'm sorry don't find a road!").

searching:-

```
open(State,Pointer),
retract(open(State,Pointer)),
closed(No,_,_),No2=No+1,
asserta(closed(No2,State,Pointer)),
!,step4(No2,State).
```

searching:- assert(fail_).

step4(_State):- mark(End),equal(State,End).

step4(No,State):- step56(No,State),!,fail.

step56(No,StateX):-

```
rule(StateX,StateY),
not(open(StateY,_)),
not(closed(_StateY,_)),
assertz(open(StateY,No)),
fail.
```

step56(_):- !.

equal(X,X).

repeat.

repeat:- repeat.

resulting(N):- closed(N,X,M),asserta(res(X)),resulting(M).

resulting(_):- res(X),write(X),nl,fail.

resulting(_):- !.

rule(X,Y):- road(X,Y).

road(v1,v2).

road(v2,v5).road(v2,v3).road(v2,v1).

road(v5,v8).road(v5,v2).road(v5,v4).

road(v4,v7).road(v4,v5).

road(v7,v8).road(v7,v4).

road(v8,v9).road(v8,v5).road(v8,v7).

road(v3,v6).road(v3,v2).

road(v6,v3).

road(v9,v8).

运行结果截图：

The screenshot shows the Prolog editor with the same code as before. The message window now displays the following output:

```

open
closed
closed
Compilation successful

```

The status bar at the bottom of the window shows the following shortcuts:

- F2-Save
- F3-Load
- F6-Switch
- F9-Compile
- Alt-X-Exit

The screenshot shows a Prolog IDE window titled "C:\DOCUME~1\ADMINI~1\桌面\课件PP~1\PROLOG.EXE". The interface includes a menu bar (File, Edit, Run, Compile, Options, Setup) and a toolbar. The main editor area contains the following code:

```

Line 1 Col 1 C:\DOCUME~1\ADMINI~1\桌面\课件PP~1\PROLOG.EXE
DOMAINS
state=symbol
DATABASE-mydatabase
open<state,integer>
closed<integer,state,integer>
res<state>
mark<state>
fail_
PREDICATES
solve
search<state,state>
result
searching
step4<integer,state>

```

The message window at the bottom left displays the following text:

```

open
closed
closed
Compilation successful

```

The status bar at the bottom shows keyboard shortcuts: F2-Save, F3-Load, F6-Switch, F9-Compile, and Alt-X-Exit.