**实验三报告 A\*算法实验I**

班级: 姓名： 学号：

1. **实验目的**

熟悉和掌握启发式搜索的定义、估价函数和算法过程，并利用A\*算法求解N数码难题，理解求解流程和搜索顺序。

1. **实验原理**

A\*算法是一种启发式图搜索算法，其特点在于对估价函数的定义上。对于一般的启发式图搜索，总是选择估价函数*f*值最小的节点作为扩展节点。因此，*f*是根据需要找到一条最小代价路径的观点来估算节点的，所以，可考虑每个节点*n*的估价函数值为两个分量：从起始节点到节点*n*的实际代价以及从节点*n*到达目标节点的估价代价。

1. **实验内容及要求**

1.参考A\*算法核心代码，以8数码问题为例实现A\*算法的求解程序（编程语言不限），要求设计两种不同的估价函数。

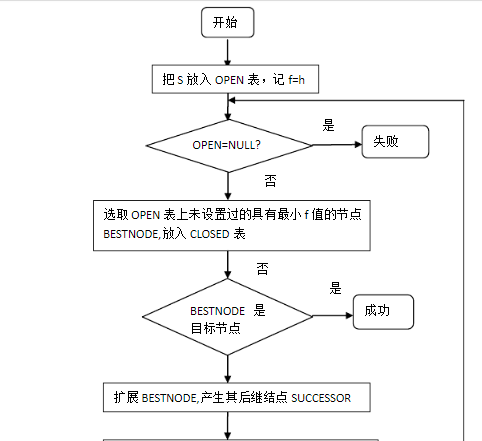
2.在求解8数码问题的A\*算法程序中，设置相同的初始状态和目标状态，针对不同的估价函数，求得问题的解，并比较它们对搜索算法性能的影响，包括扩展节点数、生成节点数等。

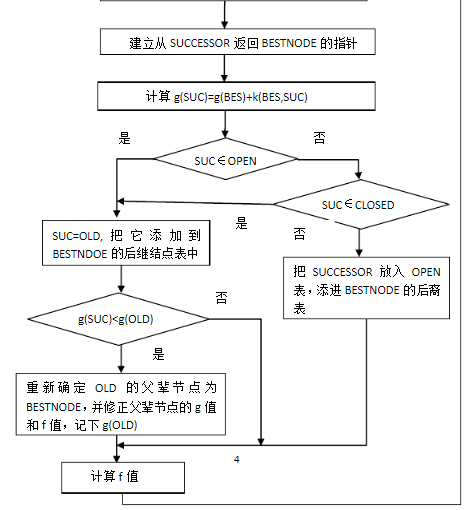
3.对于8数码问题，设置与上述2相同的初始状态和目标状态，用宽度优先搜索算法（即令估计代价h(n)＝0的A\*算法）求得问题的解，以及搜索过程中的扩展节点数、生成节点数。

4.上交源程序。

**四、实验结果（根据实验报告要求）**

1.A\*算法求解8数码问题的流程图





1. 在求解8数码问题的A\*算法程序中，设置相同的初始状态和目标状态，针对不同的估价函数，求得问题的解，并比较它们对搜索算法性能的影响，包括扩展节点数、生成节点数等。

①int calw(string s)//计算该状态的不在位数h(n)

{

int re=0;

for(int i=0;i<9;i++) if(s[i]!=t[i]) re++;

return re;

}

②int calw(string s)//计算该状态的不在位数h(n)

{

int re=0,i;

int ss[9][2];

for(i=0;i<9;i++) {

ss[s[i]-48][0]=i/3;

ss[s[i]-48][1]=i%3;

}

for(i=0;i<9;i++) {

re+=(abs(ss[i][0]-source[i][0])+abs(ss[i][1]-source[i][1]));

}

return re;

}

③int calw(string s){//计算该状态的不在位数h(n)

return 0;//宽度优先

}

1. 根据宽度优先搜索算法和A\*算法，分析启发式搜索的特点。

启发式搜索算法使得搜索的效率成倍提高。而不同的启发式搜索算法差异也较大。总之启发式搜索算法是由h(n)决定的，好的估价函数将决定算法性能的好坏。

**五、实验总结**

通过这次实验，使我对启发式搜索算法有了更进一步的理解，特别是估计函

数h(n)所起到的巨大重用。一个好的估计函数对于启发式搜索算法来说是十分关键的。