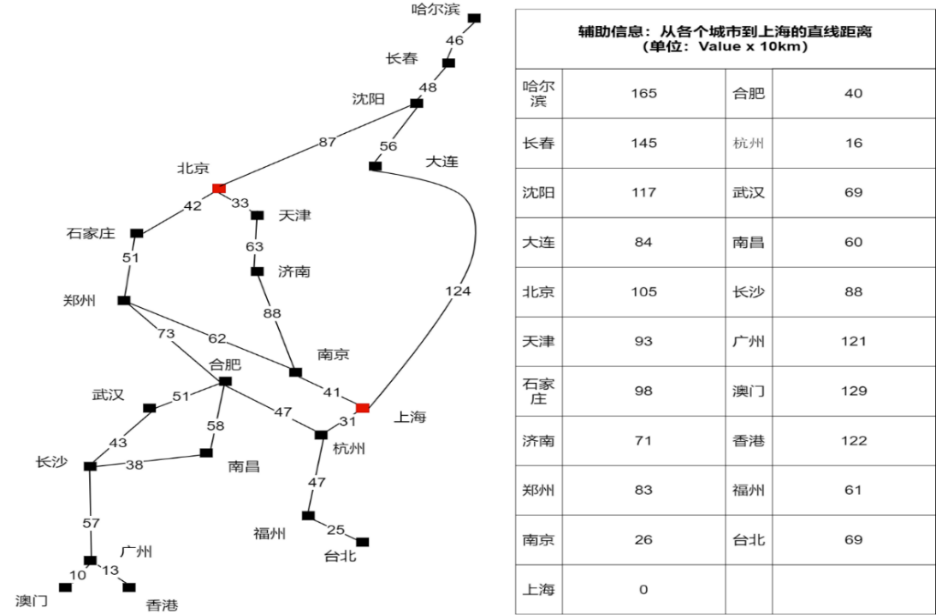
**A\*搜索：难**

As shown in the figure below, it is a major city in eastern China. It is now preparing to go from Beijing to Shanghai. The path map is on the left side of the figure above. The number on the side between the two points represents the cost between the two points. The heuristic information is from each the straight-line distance from the city to Shanghai, using the A\* algorithm to find the best path and calculate the minimum cost, list the calculation process and draw the search tree.

PS: Evaluation function f(n) = cost function g(n) + heuristic function h(n)



解：

f(北京)=0+105=105

选择北京节点，扩展天津、石家庄、沈阳。

f(天津)=33+93=126

f(石家庄)=42+98=140

f(沈阳)=87+117=204

选择天津节点，扩展济南。

f(济南)=(33+63)+71=167

选择石家庄节点，扩展郑州。

f(郑州)=(42+51)+83=176

选择济南节点，扩展南京1。

f(南京1)=(33+63+88)+26=210

选择郑州节点，扩展南京2与合肥节点。

f(南京2)=(42+51+62)+26=181

f(合肥)=(42+51+73)+40=206

选择南京2节点，扩展上海。

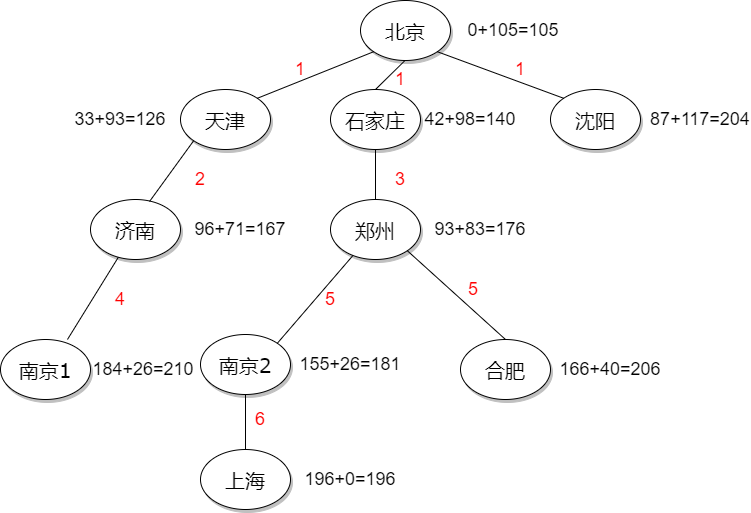
f(上海)=(42+51+62+41)+0=196

选择上海节点，得到目标节点。

最优路径为 北京->石家庄->郑州->南京2->上海

最小代价为196。

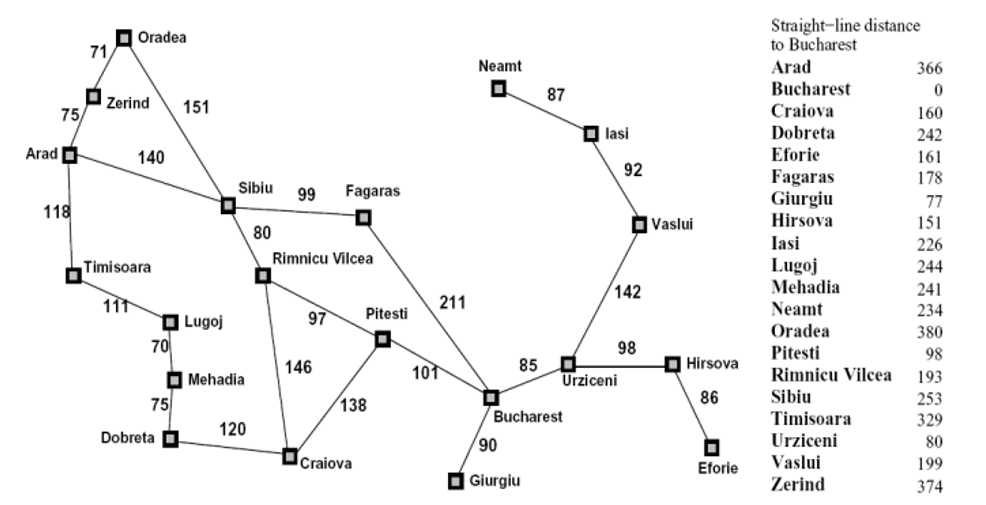
搜索树如下所示：



**A\*搜索：中等**

The following figure shows the state space diagram of the Romanian path finding problem. Using the A\* algorithm, find the best path from Oradea to Bucharest, keep the calculation process.

PS: Evaluation function f(n) = cost function g(n) + heuristic function h(n)



解：

O->B

f(O) = 0+380=380

选择O，扩展Z、S。

f(Z)=71+374=445

f(S)=151+253=404

选择S，扩展A、F、R。

f(A)=(151+140)+366=657

f(F)= (151+99)+178=428

f(R)= (151+80)+193=424

选择R，扩展C、P。

f(C)= (151+80+146)+160=537

f(P)= (151+80+97)+98=426

选择P，扩展B、C1。

f(B)= (151+80+97+101)+0=429

f(C1)= (151+80+97+138)+160=626

选择F，扩展B2。

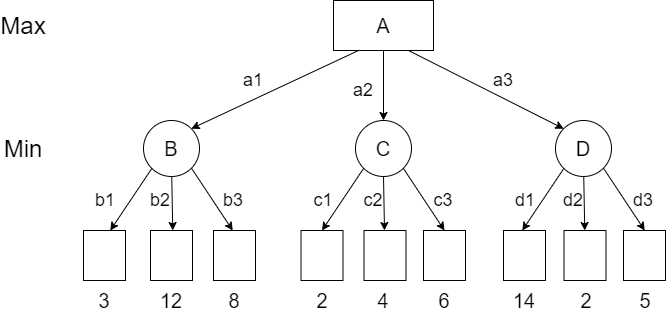
f(F)= (151+99+211)+0=461

选择B，则可得最优路径为：O->S>R>P>B。最小代价为429。

Alpha-beta剪枝：**简单**

In the max-min game confrontation, max wants to get the maximum value, and min wants to get the minimum value to fight against max, so as not to let max get the maximum value. The figure below shows the max-min game tree.

* Determine the value of A, B, C, D according to the game tree；
* use the α-β pruning algorithm to determine the branches that should be pruned；
* The reasons for pruning these branches and the significance of the pruning algorithm。



答：

A=3、B=3、C=2、D=2；

被剪的枝为：c2、c3、d3。

剪枝的作用：min在探索b1、b2、b3分支后，取最小值为3，当前max所能得到的最大值α=3，当min再探索其它分支，遇到比α小的值时，证明该分支提供的值将比max所能获取到的值小，再探索本节点的其余分支将没有意义，故可以进行剪枝，来减少时间消耗。剪枝操作在博弈搜索过程中，能够剪掉博弈树的部分子树，从而允许程序向更深处进行搜索，故可以得到更好的评估。