**八皇后问题算法**

1. 解决方案

从第一行开始，按照冲突规则过滤出每行可选的位置点集合，取一点

如果遇到死解，即当前行没有符合条件的可选点，则回退到上一行，重新取解

直到第八行取到了符合条件的安全位置，求解成功。

二．python源代码

**import** random  
**import** math  
  
  
**def** initial\_chessboard(d):  
 chessboard = [[(x+1,y+1)**for** y **in** range(d)]**for** x **in** range(d)]  
 **return** chessboard  
  
  
**def** filter\_and\_pick\_place(chessboard,rowIndex,placePicked,exclusions):  
 row=chessboard[rowIndex]  
 alternative=[]  
 **if** len(placePicked)!=0:  
 **for** column **in** row:  
 available=**True  
 for** item **in** placePicked:  
 **if** column[1]==item[1] **or** column[0]+column[1]==item[0]+item[1] **or** \  
 column[0]-column[1]==item[0]-item[1] **or** column **in** exclusions:  
 available=**False  
 if** available:  
 alternative.append(column)  
 **else**:  
 alternative=row  
  
 **if** len(alternative)==0:  
 **return** 0  
 **else**:  
 randonIndex=math.floor(len(alternative)\*random.random())  
 pick=alternative(randonIndex)  
 placePicked.append(pick)  
 **return** 1  
  
  
**def** generate\_figure(positions):  
 figureSring=**''  
 for** row **in** range(8):  
 **for** col **in** range(8):  
 **if**(row+1,col+1)**in** positions:  
 figureSring+=**'x'  
 else**:  
 figureSring+=**'-'** figureSring+=**'\n'  
 return** figureSring  
  
  
**if** \_\_name\_\_ == **'\_main\_'**:  
 chessboard=initial\_chessboard(8)  
 placePicked = []  
 exclusions=[[] **for** i **in** range(8)]  
 row =0  
 **while** row<8:  
 success=filter\_and\_pick\_place(chessboard,row,placePicked,exclusions[row])  
 **if** success==1:  
 row+=1  
 **else**:  
 row-=1  
 exclusions[row].append(placePicked.pop())  
 **if** row<7:  
 exclusions[row+1]=[]  
 print(generate\_figure(placePicked))

三．运行结果：

