

# 文本复制检测报告单(全文对照)

№:ADBD2017R\_2011092822244820170517230257509370892414

检测时间:2017-05-17 23:02:57

检测文献: 50060943843028650谭鹏

作者: 谭鹏

检测范围: 中国学术期刊网络出版总库

中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库

中国重要会议论文全文数据库

中国重要报纸全文数据库

中国专利全文数据库

互联网资源(包含贴吧等论坛资源)

英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)

港澳台学术文献库

优先出版文献库

互联网文档资源

图书资源

CNKI大成编客-原创作品库

个人比对库

时间范围: 1900-01-01至2017-05-17

## 检测结果

总文字复制比: **16.8%**

跨语言检测结果: -

去除引用文献复制比: **10.6%**

去除本人已发表文献复制比: **16.8%**

单篇最大文字复制比: **4.4%** (A\*算法寻找路径)

重复字数: [2353]

总段落数: [2]

总字数: [14033]

疑似段落数: [2]

单篇最大重复字数: [623]

前部重合字数: [447]

疑似段落最大重合字数: [1896]

后部重合字数: [1906]

疑似段落最小重合字数: [457]



指标: ☐ 疑似剽窃观点 ☒ 疑似剽窃文字表述 ☐ 疑似自我剽窃 ☐ 一稿多投 ☐ 疑似整体剽窃 ☐ 过度引用 ☐ 重复发表

表格: 0 脚注与尾注: 2

19.6% (1896) 50060943843028650谭鹏\_第1部分 (总9670字)

10.5% (457) 50060943843028650谭鹏\_第2部分 (总4363字)



(注释: 无问题部分 文字复制比部分 引用部分)

## 1. 50060943843028650谭鹏\_第1部分

总字数: 9670

相似文献列表 文字复制比: 19.6%(1896) 疑似剽窃观点: (0)

1	基于漏斗的实时VBR视频最短路径平滑算法 袁俊杰;徐小良; - 《计算机系统应用》 - 2010-07-15	5.1% (489) 是否引证: 是
2	计算机游戏设计原理以及游戏引擎的设计思想 穆俊; - 《硅谷》 - 2014-02-08	3.9% (374) 是否引证: 是
3	A*算法寻找路径 潘海波; - 《黑龙江科技信息》 - 2009-06-25	3.8% (366) 是否引证: 否
4	移动Flash流媒体关键技术的研究 袁俊杰(导师: 徐小良) - 《杭州电子科技大学硕士论文》 - 2009-12-01	3.3% (322) 是否引证: 否
5	基于BREW的手机游戏的研究和实现	3.1% (296)

	孙亭南(导师：邵时;潘荫荣) - 《华东师范大学硕士论文》 - 2006-10-01	是否引证：否
6	基于VR技术的学生群体紧急疏散模拟研究 江湉湉(导师：张际平) - 《华东师范大学硕士论文》 - 2010-09-01	3.0% ( 293 ) 是否引证：否
7	基于XNA的游戏设计与实现 黄进(导师：宋成;马迪芳) - 《北京交通大学硕士论文》 - 2011-06-01	3.0% ( 286 ) 是否引证：否
8	3DXML标准在协同装配中的研究与应用 曹翔(导师：黄志球) - 《南京航空航天大学硕士论文》 - 2007-01-01	2.6% ( 251 ) 是否引证：否
9	直线的DDA算法的C实现 ( opengl实现 ) - zhouyelihua - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2013	2.3% ( 218 ) 是否引证：否
10	人工智能寻路算法在电子游戏中的研究和应用 詹海波(导师：沈刚) - 《华中科技大学硕士论文》 - 2006-10-01	2.2% ( 212 ) 是否引证：否
11	人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012	2.2% ( 212 ) 是否引证：否
12	深入浅出A*算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015	2.1% ( 206 ) 是否引证：否
13	A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - 2012	2.0% ( 194 ) 是否引证：否
14	A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015	2.0% ( 194 ) 是否引证：否
15	A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2011	1.9% ( 187 ) 是否引证：否
16	AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2016	1.9% ( 187 ) 是否引证：否
17	3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012	1.8% ( 176 ) 是否引证：否
18	蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012	1.8% ( 176 ) 是否引证：否
19	基于HTML5与Node.js的游戏引擎的设计与开发 徐佳宾(导师：曹士珂) - 《南京邮电大学硕士论文》 - 2016-11-18	1.8% ( 176 ) 是否引证：否
20	群体动画中自主智能体的行为控制 杨延(导师：张建新) - 《电子科技大学硕士论文》 - 2009-06-01	1.7% ( 163 ) 是否引证：否
21	一款基于ARPG的“仙剑”手机游戏的设计与实现 蔡丽鸳(导师：孙涌) - 《苏州大学硕士论文》 - 2016-09-01	1.1% ( 109 ) 是否引证：否

	原文内容	相似内容来源
1	<p>此处有 375 字相似</p> <p>smoothing algorithm ,shaking, A * algorithm</p> <p>一、引言</p> <p>1.1 背景</p> <p>在</p> <p>计算机游戏中，模拟游戏 ( Sim )、射击游戏 ( shooter ) 角色扮演游戏 ( RPG ) 是较早并且具有较强代表性的计算机游戏，这些游戏的设计、产生和应用为计算机游戏整体设计水平的提升奠定的良好的基础。在这些早期的游戏设计过程中，游戏设计者通过让玩家利用一种非结构化的方式来进行游戏体验，这种游戏体验往往具有较为明确的目标，例如角色扮演游戏则是以剧情的进展、而射击游戏是以关卡的演进为明确的目标，但是在明确的设计结构下游戏玩家的动作是随意的，即在这种游戏设计原理下游戏玩家能够进行具有探索性的游戏方式，这种游戏方式的存在极大的提升了计算机游戏的可玩性并且为接下来计算机游戏的设计水平提升提供了重要助力。即游戏玩家通过自主探索在实际上可以有效反应出游戏设计的基本结构，这种结构在提升游戏系统明确性的同时可以较为集中的反应游戏设计</p>	<p>计算机游戏设计原理以及游戏引擎的设计思想 穆俊;-《硅谷》- 2014-02-08 ( 是否引证：是 )</p> <p>1.狂热的术语海洋中,即对计算机游戏设计原理解释出现了多样化的情况,并且这种多样化的解释最终并没有形成较为统一的理解。在计算机游戏中,模拟游戏 (Sim)、射击游戏(shooter)角色扮演游戏(RPG)是较早并且具有较强代表性的计算机游戏,这些游戏的设计、产生和应用为计算机游戏整体设计水平的提升奠定的良好的基础。在这些早期的游戏设计过程中,游戏设计者通过让玩家利用一种非结构化的方式来进行游戏体验,这种游戏体验往往具有较为明确的目标,例如角色扮演游戏则是以剧情的进展、而射击游戏是以关卡的演进为明确的目标,但是在明确的设计结构下游戏玩家的动作是随意的,即在这种游戏设计原理下游戏玩家能够进行具有探索性的游戏方式,这种游戏方式的存在极大的提升了计算机游戏的可玩性并且为接下来计算机游戏的设计水平提升提供了重要助力。即游戏玩家通过自主探索在实际上可以有效反应出游戏设计的基本结构,这种结构在提升游戏系统明确性的同时可以较为集中的反应游戏设计接下来的进步方向。在这之后,计算机游戏的设计从追寻某个特点</p>

	<p>接下来的进步方向。</p> <p>[1]</p> <p>随着个人电脑性能的提升、普通用户智能手机的普及，使得现如今人们所持有的设备的性能越来越高，3D游戏所需要的额外</p>	<p>的目标,例如角色扮演的剧情补完和射击游戏的关卡全通进入到了新的领域,即互动。这一</p>
2	<p>此处有 213 字相似</p> <p>在的点设为新的起点来构建漏斗，重复进行。但是这个算法是基于边界和点的。在某些情况下会效率比较低下，不适用于某些游戏。</p> <p>如图1所示，<math>O(t)</math>和<math>U(t)</math>构成的区域可以看作是一个单调多边形。我们用<math>\Pi(s,w)</math>来表示多边形内点 <math>s</math>至点 <math>w</math> 的最短路径，那么 <math>\Pi(s,e)</math> 即为所求最短路径。显然，<math>\Pi(s,e)</math>只可能与多边形的凹点(例如 <math>v</math>)相交，不可能与凸点(例如 <math>z</math>)相交。因此，计算最短路径只需要考虑凹点。假设已知 <math>\Pi(u,s) = [u,s]</math> 及 <math>\Pi(s,w) = [s,v,w]</math>，那么漏斗就是由 <math>\Pi(u,s)</math> 以及 <math>\Pi(s,w)</math> 构成的 V形结构，用 <math>F_{uw}</math> 表示，而 <math>s</math> (V形底端)为漏斗的底。<math>F_{uw}</math>是由最初漏斗经过数次更新后生成的，最初的漏斗由起始点</p>	<p>基于漏斗的实时VBR视频最短路径平滑算法 袁俊杰;徐小良;-《计算机系统应用》-2010-07-15 (是否引证：是)</p> <p>1.频数据⑤根据路径特征向后滑动窗口,返回第①步。 ⑥按照求得的最短路径传输窗口内的数据,程序结束。</p> <p>3.2基于漏斗求解最短路径如图2所示,<math>O(t)</math>和<math>U(t)</math>构成的区域可以看作是一个单调多边形。我们用<math>\Pi(s,w)</math>来表示多边形内点<math>s</math>至点<math>w</math>的最短路径,那么<math>\Pi(s,e)</math>即为所求最短路径。显然,<math>\Pi(s,e)</math>只可能与多边形的凹点(例如<math>v</math>)相交,不可能与凸点(例如<math>z</math>)相交。因此,计算最短路径只需要考虑凹点。假设已知<math>\Pi(u,s)=[u,s]</math>及<math>\Pi(s,w)=[s,v,w]</math>,那么漏斗就是由<math>\Pi(u,s)</math>以及<math>\Pi(s,w)</math>构成的</p> <p><math>VU(t) \leq S(t) \leq O(t)</math> <math>U(t) = U(t) + b_i i</math> <math>i_{max}</math></p>
3	<p>此处有 341 字相似</p> <p>及 <math>\Pi(s,w) = [s,v,w]</math>，那么漏斗就是由 <math>\Pi(u,s)</math> 以及 <math>\Pi(s,w)</math> 构成的 V形结构，用 <math>F_{uw}</math> 表示，而 <math>s</math> (V形底端)为漏斗的底。<math>F_{uw}</math>是由最初漏斗经过数次更新后生成的，最初的漏斗由起始点 <math>s</math> 与 <math>O(t)</math>、<math>U(t)</math> 的第一个凹点的连线构成的，如图中虚实线所示。</p> <p>图1</p> <p>在现有漏斗 <math>F_{uw}</math> 的基础上，计算出漏斗底至下一凹点 <math>x</math> 的最短路径 <math>\Pi(s,x)</math>，那么漏斗就更新为 <math>F_{ux} = \Pi(u,s) + \Pi(s,x)</math>，依次类推计算后续凹点直至终点 <math>e</math>，即可得到 <math>F_{*e}</math>，而 <math>F_{*e}</math> 的下半部分即为所求的 <math>\Pi(s,e)</math>，在 <math>F_{uw}</math> 的基础上，计算 <math>\Pi(s,x)</math> 的方法如下。首先，如果 <math>s</math> 到 <math>x</math> 的直线不与 <math>F_{uw}</math> 相交，那么 <math>\Pi(s,x)</math> 就是直线 <math>sx</math>，否则，在 <math>F_{ux}</math> 中寻找一个点 <math>v_i</math>，使得 <math>vix</math> 与 <math>F_{ux}</math>相切。用公式可以表示为：</p> <p><math>tg(v_{i-1}v_i) \geq tg(vix) \geq tg(vivi+1)</math></p> <p>其中 <math>tg</math> 表示线段的斜率，<math>v_{i-1}</math> <math>v_i</math> 以及 <math>v_{i+1}</math></p>	<p>移动Flash流媒体关键技术的研究 袁俊杰 -《杭州电子科技大学硕士论文》-2009-12-01 (是否引证：否)</p> <p>1.，那么漏斗就更新为 <math>F_{ux} = \Pi(u,s) + \Pi(s,x)</math>，依次类推计算后续凹点直至终点 <math>e</math>，即可得 <math>\Pi(s,e)</math>。<math>F_{uw}</math> 是由最初漏斗经过数次更新后生成的，最初漏斗由起始点 <math>s</math> 与第一个凹点对连线构成的，如图 5.3 中虚实线所示。图 5.3 计算最短路径示意图 在 <math>F_{uw}</math> 的基础上，计算 <math>\Pi(s,x)</math>非常方便。首先，如果</p> <p>2.法是遍历多边形上下凹点对的过程。在遍历的过程中需要维护一个漏斗，并借助漏斗求解最短路径。如图 5.3 所示，假设现有漏斗 <math>F_{uw}</math>，计算出漏斗底至下一凹点 <math>x</math> 的最短路径 <math>\Pi(s,x)</math>，那么漏斗就更新为 <math>F_{ux} = \Pi(u,s) + \Pi(s,x)</math>，依次类推计算后续凹点直至终点 <math>e</math>，即可得 <math>\Pi(s,e)</math>。<math>F_{uw}</math> 是由最初漏斗经过数次更新后生成的，最初漏斗由起始点 <math>s</math> 与第一个凹点对连线构成的，如</p> <p>3.与第一个凹点对连线构成的，如图 5.3 中虚实线所示。图 5.3 计算最短路径示意图 在 <math>F_{uw}</math> 的基础上，计算 <math>\Pi(s,x)</math>非常方便。首先，如果 <math>s</math> 到 <math>x</math> 的直线不与 <math>F_{uw}</math> 相交，那么 <math>\Pi(s,x)</math>就是直线 <math>sx</math>。否则，在 <math>F_{uw}</math> 中寻找一个点 <math>v_i</math>，使得 <math>vix</math> 与 <math>F_{uw}</math>相切。用公式可以表示为：</p> <p><math>tg(v_i v_{i+1}) \geq tg(vix) \geq tg(vivi+1)</math></p> <p>基于漏斗的实时VBR视频最短路径平滑算法 袁俊杰;徐小良;-《计算机系统应用》-2010-07-15 (是否引证：是)</p> <p>1.<math>\geq \min(y)K M22i ii1i1(x x)/K(y y)/M==\sum? \leq \sum? i</math> <math>i x=\sum x, y=\sum y</math>形结构,用<math>F_{uw}</math>表示,而<math>s</math>(V形底端)为漏斗的底。<math>F_{uw}</math>是由最初漏斗经过数次更新后生成的,最初的漏斗由起始点<math>s</math>与<math>O(t)</math>、<math>U(t)</math>的第一个凹点的连线构成的,如图</p>

		<p>中虚实线所示。图2漏斗在现有漏斗Fuw的基础上,计算出漏斗底至下一凹点x的最短路径<math>\Pi(s,x)</math>,那么漏斗就更新为<math>Fux=\Pi(u,s)+\Pi(s,x)</math>,依次类推计算后续凹点直至终点e,即可得到<math>F^*e</math>,而<math>F^*e</math>的下半部分即为所求的<math>\Pi(s,e)</math>。在Fuw的基础上,计算<math>\Pi(s,x)</math>的方法如下。首先,如果s到x的直线不与Fuw相交,那么<math>\Pi(s,x)</math>就是直线sx。否则,在Fuw中寻找一个点<math>v_i</math>,使得<math>vix</math>与Fuw相切。用公式可以表示为:(6)其中tg表示线段的斜率</p>
4	<p>此处有 47 字相似</p> <p>(vivi+1) 其中 tg 表示线段的斜率, <math>v_{i-1}v_i</math>以及 <math>v_iv_{i+1}</math> 为 Fuw 中<math>v_i</math> 的 前后线段。找到点 <math>v_i</math> 后连接<math>vix</math>, 那么 <math>\Pi(s,x) = [s, \dots v_i, x][2]</math>。</p> <p>图2 以图2来说明漏斗的过程: 1. 起点与两边边界最开始的点连线, 这两条线构成了一个漏斗。 2. 分别测试两边边</p>	<p>移动Flash流媒体关键技术的研究 袁俊杰 - 《杭州电子科技大学硕士论文》 - 2009-12-01 ( 是否引证: 否 )</p> <p>1. (5.1) 其中 tg 表示线段的斜率, <math>v_{i-1}v_i</math>、<math>v_iv_{i+1}</math> 为 Fuw中处于点 <math>v_i</math> 前后的线段。找到点 <math>v_i</math> 后连接<math>vix</math>, 那么 <math>\Pi(s,x) = [s, \dots v_i, x]</math>。在最短路径求解过程中还需要考虑如何更新漏斗。漏斗的更新存在两种情况, 第一种如之前所讲, 只需要对漏斗的左</p>
5	<p>此处有 112 字相似</p> <p>际项目中。 二、相关算法 2.1 A*算法 A*算法原本是一种求解最短路径直接搜索方法, 是一种启发式搜索算法。 公式表示为: <math>f(n) = g(n) + h(n)</math> 其中, <math>f(n)</math> 是从初始状态经由状态n到目标状态的代价估计, <math>g(n)</math> 是在状态空间中从初始状态到状态n的实际代价, <math>h(n)</math> 是从状态n到目标状态的最佳路径的估计代价。</p> <p>A*算法的具体过程为: 1. 把起始格添加到开启列表。 2. 寻找开启列表中F值最低的格子, 这就是当前格。将这个格子加入</p>	<p>A*算法寻找路径 潘海波; - 《黑龙江科技信息》 - 2009-06-25 ( 是否引证: 否 )</p> <p>1.A*(A-Star)算法是一种静态路网中求解最短路最有效的方法。公式表示为:<math>f(n)=g(n)+h(n)</math>,其中<math>f(n)</math>是节点n从初始点到目标点的估价函数,<math>g(n)</math>是在状态空间中从初始节点到n节点的实际代价,<math>h(n)</math>是从n到目标节点最佳路径的估计代价。保证找到最短路径(最优解的)条件,关键在于估价函数<math>h(n)</math>的选取:估价值<math>h(n) \leq n</math>到目标节点的距离实际值,这种情况下</p> <p>一款基于ARPG的“仙剑”手机游戏的设计与实现 蔡丽鸳 - 《苏州大学硕士论文》 - 2016-09-01 ( 是否引证: 否 )</p> <p>1.问题, 同时也是很多其他问题的最常用启发式搜索算法。之后涌现了更多的预处理算法 ( CH, ALT, HL等等 )。函数公式表示为: <math>f(n)=g(n)+h(n)</math> 公式3-1其中: <math>f(n)</math>是从初始状态经由状态n到目标状态的代价估计; <math>g(n)</math>是在状态空间中从初始状态到状态n的实际代价; <math>h(n)</math>是从状态n到目标状态的最佳路径的估计代价。根据这个函数可以计算出每个节点的代价, 每次进行搜索时, 通过这个估价函数<math>f(n)</math>对下一步能够达到的每一个点进行评价,</p> <p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证: 否 )</p> <p>1.是用估价函数表示的, 如: <math>f(n) = g(n) + h(n)</math> 其中<math>f(n)</math>是节点 n 的估价函数, <math>g(n)</math>实在状态空间中从初始节点到 n 节点的实际代价, <math>h(n)</math>是从 n 到目标节点最佳路径的估计代价。在这里主要是 <math>h(n)</math>体现了搜索的启发信息, 因为 <math>g(n)</math>是已知的。如果说详细点, <math>g(n)</math>代表了搜索</p> <p>人工智能寻路算法在电子游戏中的研究和应用 -</p>



		<p>docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证 : 否 )</p> <p>1.是如何表示的。启发中的估价是用估价函数表示的 , 如 : <math>f(n) = g(n) + h(n)</math> 其中<math>f(n)</math>是节点<math>n</math>的估价函数 , <math>g(n)</math>实在状态空间中从初始节点到<math>n</math>节点的实际代价 , <math>h(n)</math>是从<math>n</math>到目标节点最佳路径的估计代价。在这里主要是<math>h(n)</math>体现了搜索的启发信息 , 因为<math>g(n)</math>是已知的。如果说详细点 , <math>g(n)</math>代表了搜索的广度的优先趋势。但是当</p>
6	<p>此处有 178 字相似</p> <p>n) 是在状态空间中从初始状态到状态<math>n</math>的实际代价 , <math>h(n)</math> 是从状态<math>n</math>到目标状态的最佳路径的估计代价。</p> <p>A*算法的具体过程为 :</p> <p>1 . 把起始格添加到开启列表。</p> <p>2 . 寻找开启列表中F值最低的格子 , 这就是当前格。将这个格子加入到关闭列表。</p> <p>3. 对于当前格相邻的每一个格子 , 如果这一格不可通过或者已经在关闭列表中 , 那么就略过这一格 ; 如果这一格不在开启列表中 , 把这一格添加进去开启列表 , 并且把当前格作为这一格的父节点 , 记录这一格的F,G,和H的值 ; 如果这一格已经在开启列表中 , 则检查这一格的G值是否更低 , 如果是 , 就把这一格的父节点改成当前格 , 并且重新计算这一格的G和F值 ; 如果你把目标格添加进了关</p>	<p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p> <p>1.估算 , 而非实际值 , 这也是此方法被称为启发式的原因。 2.A*方法过程 (1)把起始格添加到开启列表。(2)重复如下工作: a)寻找开启列表中F值最低的格子。称它为当前格。 b)把它切换到关闭列表。 c)对相邻的8格中的每一个。如果它不可通过或者已经在关闭列表中 , 略过它。反之 , ①如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的F , G , 和H值;②如果它已经在开启列表中 , 用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样 ,</p> <p>A*算法寻找路径 潘海波; - 《黑龙江科技信息》 - 2009-06-25 ( 是否引证 : 否 )</p> <p>1.。另一方面,如果新的G值更低,那就把相邻方格的父节点改为目前选中的方格。最后,重新计算F和G的值。 A*方法总结:(1)把起始格添加到开启列表。(2)重复如下工作:a.寻找开启列表中F值最低的格子。称它为当前格。b.把它切换到关闭列表。c.对相邻的8格中的每一个:如果它不可通过或者已经在关闭列表中,略过它。反之如下。如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一</p> <p>基于VR技术的学生群体紧急疏散模拟研究 江湉湉 - 《华东师范大学硕士论文》 - 2010-09-01 ( 是否引证 : 否 )</p> <p>1. 341把起始格添加到开启列表。·—2重复如下工作: a)寻找开启列表中F值最低的格子 , 称为当前格。_ b)把它切换到关闭列表。_ c)对相邻的8格中的每一个: 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 。如果它不在开启列表中 , 把它添加进去。把当前格作设这一格的父节点。记录这一格的F、G和H值 , ·如果它已经在开启列表中 , 用G值为参考检查新的路径是否更好。 更低的G值意味着更好的路径。如果是这样 , 就把这一格的父节点</p> <p>A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - ( 是否引证 : 否 )</p> <p>1.把起始格添加到开启列表。 2 , 重复如下工作 : a) 寻找开启列表中F值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个 ? * 如果它不可通过或者已经在关闭列表中 , 略过它。反</p>

		<p>之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p>
		<p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )</p> <p>1.把起始格添加到开启列表。2，重复如下的工作： a) 寻找开启列表中F值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个？ * 如果它不可通过或者已经在关闭列表中，略过它。反之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p>
		<p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证：否 )</p> <p>1.把起始格添加到开启列表。2，重复如下的工作： a) 寻找开启列表中F值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个？ * 如果它不可通过或者已经在关闭列表中，略过它。反之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p>
		<p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》 - 2011-06-01 ( 是否引证：否 )</p> <p>1.作: a)寻找开启列表中F值最低的格子。我们称它为当前格。 b)把它一切换到关闭列表。 C)对相邻的8格中的每一个格子进行检测:如果它不可通过或者已经在 关闭列表中，略过它。如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F，G，和日值。如果它己 经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值 意味着更好的路径。如果是这样，就把这一格的父节</p>
		<p>深入浅出A&amp;#x2d;Star算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015-7-24 0:28:46 ( 是否引证：否 )</p> <p>1.始格添加到开启列表。 ,2、 重复如下的工作,a) 寻找开启列表中F值最低的格子。我们称它为当前格。 ,b) 把它切换到关闭列表。 ,c) 对相邻的8格中的每一个,如果它不可通过或者已经在关闭列表中,略过它。反之如下。 ,如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 ,如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。 如果是这样,就把这一格</p>

		<p>的父节点改成当前格,并且重新计算这一</p>
		<p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-16 12:07:06 ( 是否引证 : 否 )</p>
		<p>1.作 : a) 寻找开启列表中 F 值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的 8 格中的每一个 ? * 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 * 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F,G,和 H 值。 * 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样 , 就把这一格的父节点改成当前格 ,</p>
		<p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-19 23:58:36 ( 是否引证 : 否 )</p>
		<p>1.的工作 : a) 寻找开启列表中 F 值最低的格子。我们称它为当前格。 b) 把它切换到 关闭列表。 c) 对相邻的 8 格中的每一个 ? 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F,G,和 H 值。 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样 , 就把这一格的父节点改成当前格 , 并且重</p>
		<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证 : 否 )</p>
		<p>1.把起始格添加到开启列表。 2. 重复如下的工作,a) 寻找开启列表中F值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个,1. 如果它不可通过或者已经在关闭列表中,略过它。反之如下。 2. 如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 3. 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这</p>
		<p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p>
		<p>1.列表。 ( c ) 对相邻的 8 格中的每一格进行如下操作 : * 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 * 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F,G,和 H 值。 * 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样 , 就把这一格的父节点</p>
		<p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学硕士论文》 - 2009-06-01 ( 是否引证 : 否 )</p>
		<p>1. c) 对相邻的 8 格中的每一个格子进行如下判断 : * 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 * 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F,G,和 H 值。 *</p>

		<p>如果它已经在开启列表中，用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样，就把这一格的父节点</p> <p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证：否 )</p> <p>1.列表中值最低的格子。我们称它为当前格。（b）把它切换到关闭列表。（c）对相邻的8格中的每一格进行如下操作：* 如果它不可通过或者已经在关闭列表中，略过它。反之如下。* 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。* 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p>
7	<p>此处有 90 字相似</p> <p>，并且把当前格作为这一格的父节点，记录这一格的F,G,和H的值；如果这一格已经在开启列表中，则检查这一格的G值是否更低，</p> <p>如果是，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值；如果你把目标格添加进了关闭列表，则说明路径被找到，或者没有找到目标格，但这个时候开启列表已经空了则说明路径不存在</p> <p>则结束这一步执行4。遍历完所有邻格后，执行2。</p> <p>4. 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这</p>	<p>A*算法寻找路径 潘海波;-《黑龙江科技信息》- 2009-06-25 ( 是否引证：否 )</p> <p>1.节点。记录这一格的F,G,和H值。如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d.停止:把目标格添加进了开启列表,这时候路径被找到</p> <p>2.前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d.停止:把目标格添加进了开启列表,这时候路径被找到。没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。责任编辑:王青翠A*算法寻找路径@潘</p> <p>基于VR技术的学生群体紧急疏散模拟研究 江湉湉 -《华东师范大学硕士论文》- 2010-09-01 ( 是否引证：否 )</p> <p>1. 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。 更低的G值意味着更好的路径。如果是这样，就把这一格的父节点 改成当前格，并且重新计算这一格的G和F值。 d)停止，当：.把目标格添加进了开启列表，这时候路径被找到，或者。没有找到目标格，开启列表已经空了。这时候，路径不存在。3保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是所求的路径</p> <p>基于XNA的游戏设计与实现 黄进 -《北京交通大学硕士论文》- 2011-06-01 ( 是否引证：否 )</p> <p>1. 经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值 意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排 序，改变之后你可能需要重新对开启列表排序。</p> <p>2.表按F值排 序，改变之后你可能需要重新对开启列表</p>



	<p>排序。d)停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这</p>
	<p>深入浅出A*Star算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015-7-24 0:28:46 (是否引证:否)</p>
	<p>1.。记录这一格的F,G,和H值。 ,如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 ,d) 停止,当你把目标格添加进了开启列表,这时候</p> <p>2.并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 ,d) 停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。 ,3、保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 ,题外话 ,</p>
	<p>A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - (是否引证:否)</p>
	<p>1.记录这一格的F,G,和H值。 * 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 * 把目标格添加进了开启列表,这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 * 把目标格添加进了开启列表,这时候路径被找到,或者 * 没有找到目标格,开启列表已经空了。这时候,路径不存在。 3.保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题一</p>
	<p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - (是否引证:否)</p>
	<p>1.记录这一格的F,G,和H值。 * 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 * 把目标格添加进了关闭列表,这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 * 把目标格添加进了关闭列表,这时候路</p>

	<p>怪被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。(注解:在这篇</p>
	<p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证：否 )</p> <p>1.记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 * 把目标格添加进了开启列表，这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 * 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p>
	<p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 ( 是否引证：否 )</p> <p>1.H值;②如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果保持开启列表按F值排序，改变之后可能需要重新对开启列表排序。</p> <p>2.序，改变之后可能需要重新对开启列表排序。 d)停止当把目标格添加进了开启列表，这时候路径被找到。或者没有找到目标格，开启列表已经空了。这时候，路径不存在。(3)保存路径。从目标格开始，沿着每一格的父节点移动直到回到起</p>
	<p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证：否 )</p> <p>1.记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。( d ) 停止，当我们 * 把目标格添加进了关闭列表</p> <p>2.这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。( d ) 停止，当我们 * 把目标格添加进了关闭列表这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。( 3 ) 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是要</p>

	求的最短路径。 2.4
	<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证 : 否 )</p> <p>1.录这一格的F,G,和H值。 3. 如果它已经在开启列表中 ,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。 <b>如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。</b>如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 4. 把目标格添加进了开启列表,</p> <p>2.算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 4. 把 <b>目标格添加进了开启列表,这时候路径被找到,或者 5. 没有找到目标格,开启列表已经空了。</b>这时候 ,路径不存在。 3. 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题</p>
	<p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p> <p>1.* 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。 <b>如果是这样 , 就把这一格的父节点改成当前格 , 并且重新计算这一格的 G 和 F 值。</b>如果你保持你的开启列表按 F 值排序 , 改变之后你可能需要重新对开启列表排序。</p> <p>2.新对开启列表排序。 ( d ) 停止 , 当我们 * 把 <b>目标格添加进了关闭列表 , 这时候路径被找到 , 或者 * 没有找到目标格 , 开启列表已经空了。</b>这时候 , 路径不存在。 ( 3 ) 保存路径。从目标格开始 , 沿着每一格的父节点移动直到回到起始格。这就</p>
	<p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学硕士论文》 - 2009-06-01 ( 是否引证 : 否 )</p> <p>1. * 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。 <b>如果是这样 , 就把这一格的父节点改成当前格 , 并且重新计算这一格的 G 和 F 值。</b>如果保持你的开启列表按 F 值排序 , 改变之后可能需要重新对开启列表排序。</p> <p>2.列表按 F 值排序 , 改变之后可能需要重新对开启列表排序。 d) 停止 , 当 <b>目标格添加进了关闭列表 , 这时候路径被找到 , 或者没有找到目标格 , 开启列表已经空了。</b>这时候 , 路径不存在。 3 ) 保存路径。从目标格开始 , 沿着每一格的父节点移动直到回到起始格。这就是你</p>
	<p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-16 12:07:06 ( 是否引证 : 否 )</p> <p>1.G,和 H 值。 * 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。 更低的 G 值意味着更好的路径。 <b>如果是这样 , 就把这一格的父节点改成当前格 , 并且重新计算这一格的 G 和 F 值。</b>如果你保持你的开启列表按 F 值排序 , 改变之后你可能需要重新对开</p>

		<p>启列表排序。d) 停止，当你 47 北京</p> <p>2.启列表按 F 值排序，改变之后你可能需要重新对开启列表排序。d) 停止，当你 47 北京理工大学工程硕士学位论文 * 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3. 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。1.4 题</p> <p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-19 23:58:36 ( 是否引证：否 )</p> <p>1. F,G,和 H 值。如果它已经在开启列表中，用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的 G 和 F 值。如果你保持你的开启列表按 F 值排序，改变之后你可能需要重新对开启列表排序。d) 停止，当你 把目标</p> <p>2.的 G 和 F 值。如果你保持你的开启列表按 F 值排序，改变之后你可能需要重新对开启列表排序。d) 停止，当你 把目标格添加进了开启列表，这时候路径被找到，或者 没有找到目标格，开启列表已经空了。这时候，路径不存在。保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。题外话 离题一下，</p>
8	<p>此处有 57 字相似</p> <p>，或者没有找到目标格，但这个时候开启列表已经空了则说明路径不存在则结束这一步执行4。遍历完所有邻格后，执行2。</p> <p>4.</p> <p>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p> <p>2.2 弗洛伊德路径平滑算法</p> <p>弗洛伊德算法是解决任意两点间的最短路径的一种算法，能够正确解决有向图或存在负权的图的最短路径问题，同时也常常被用于计算有</p>	<p>基于VR技术的学生群体紧急疏散模拟研究 江湑湑 - 《华东师范大学硕士论文》 - 2010-09-01 ( 是否引证：否 )</p> <p>1.表中·d)停止，当-没有找到目标格，开启列表已经空了。这时候，路径不存在。3保存路径。从终点格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。改进后的算法将可以在具有多个终点的地图中进行最佳的路径选择，更加智能化，也更加符合在疏散模拟中的路径选择情况，使得逃离的个</p> <p>A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - ( 是否引证：否 )</p> <p>1. * 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。题外话 离题一下，见谅，值得一提的是，当你在网上或者相关论坛看到关于A*的不同的探讨，你有时会看到一些被当作A*算法的代</p> <p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-16 12:07:06 ( 是否引证：否 )</p> <p>1.* 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3. 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。1.4 题外话 值得一提的是，当你在网上或者相关论坛看到关于 A*的不同的探讨，你有时会看到一些被当作</p>



		A*算法的代码而实际上
		<p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-19 23:58:36 ( 是否引证 : 否 )</p> <p>1. 停止, 当你 把目标格添加进了开启列表, 这时候路径被找到, 或者 没有找到目标格, 开启列表已经空了。这时候, 路径不存在。 <b>保存路径。从目标格开始, 沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 题外话 离题一下, 见谅, 值得一提的是, 当你在网上或者相关论坛看到关于 A*的不同的 探讨, 你有时会看到一些被当作 A*算</p>
		<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证 : 否 )</p> <p>1. 把目标格添加进了开启列表,这时候路径被找到,或者 5. 没有找到目标格,开启列表已经空了。这时候,路径不存在。 3. <b>保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 题外话 离题一下,见谅,值得一提的是,当你在网上或者相关论坛看到关于A*的不同的探讨,你有时会看到一些被当作A*算法的代</p>
		<p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证 : 否 )</p> <p>1. * 把目标格添加进了关闭列表, 这时候路径被找到, 或者 * 没有找到目标格, 开启列表已经空了。这时候, 路径不存在。 3.<b>保存路径。从目标格开始, 沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> (注解:在这篇文章的较早版本中, 建议的做法是当目标格 ( 或节点 ) 被加入到开启列表, 而不是关闭列表的时候停止寻路。这么做会</p>
		<p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证 : 否 )</p> <p>1. * 把目标格添加进了开启列表, 这时候路径被找到, 或者 * 没有找到目标格, 开启列表已经空了。这时候, 路径不存在。 3.<b>保存路径。从目标格开始, 沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b></p>
		<p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p> <p>1.找到目标格, 开启列表已经空了。这时候, 路径不存在。(3)<b>保存路径。从目标格开始, 沿着每一格的父节点移动直到回到起始格。这就是移动的路径。</b> 3.A*实现 (1)维护开启列表: 这是</p>
		<p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学硕士论文》 - 2009-06-01 ( 是否引证 : 否 )</p> <p>1.进了关闭列表, 这时候路径被找到, 或者没有找到目标格, 开启列表已经空了。这时候, 路径不存在。 3 ) <b>保存路径。从目标格开始, 沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 在介绍完 A*算法的实现原理后, 下面具体给出 A*算法的实现代码。 60</p>

		<p>A*算法寻找路径 潘海波; - 《黑龙江科技信息》 - 2009-06-25 ( 是否引证 : 否 )</p> <p>1.排序。d.停止:把目标格添加进了开启列表,这时候路径被找到。没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。责任编辑:王青翠A*算法寻找路径@潘海波\$黑龙江旅游职业技术学院!黑龙江哈尔滨150086介绍的就是实现如何用A*算法在</p> <p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》 - 2011-06-01 ( 是否引证 : 否 )</p> <p>1.找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。3.蚁群算法 蚁群算法(A, It eoroll)optil二 izatio , 1.ACo)是一种最新</p> <p>深入浅出A*Star算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015-7-24 0:28:46 ( 是否引证 : 否 )</p> <p>1.停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。3、保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。题外话,离题一下,见谅,值得一提的是,当你在网上或者相关论坛看到关于A*的不同的探讨,你有时会看到一些被当作A*算</p> <p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p> <p>1.*没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是要求的最短路径。2.4 本章小结 本章主要介绍了电子游戏中人工智能寻路算</p> <p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证 : 否 )</p> <p>1.*把目标格添加进了关闭列表这时候路径被找到,或者*没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是要求的最短路径。2.4 本章主要介绍了电子游戏中人工智能寻路算法涉及的人工智能、图的遍历算法,最短路径搜索等关键性技术</p>
9	<p>此处有 79 字相似</p> <p>个节点。</p> <p>2.3 DDA算法</p> <p>数值微分法即DDA法(Digital Differential Analyzer),是一种基于直线的微分方程来生成直线的方法。</p> <p>算法描述:</p> <p>设(x1,y1)和(x2,y2)分别为所求直线的起点和终点坐标,由直线的微分方程得:</p>	<p>3DXML标准在协同装配中的研究与应用 曹翔 - 《南京航空航天大学硕士论文》 - 2007-01-01 ( 是否引证 : 否 )</p> <p>1.:数值微分法 ( DDA )、中点画线法和 Bresenham 算法。DDA 算法:一种基于直线的微分方程来生成直线的方法。设(x1,y1)和(x2,y2)分别为所求直线的起点和终点坐标,由直线的微分方程得: <math>xyxyydx dy</math></p> <p>直线的DDA算法的C实现 ( opengl实现 ) - zhouyelihua - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 -</p>

	<p><b>dx/dy</b></p> <p><math>= (y_2 - y_1) / (x_2 - x_1) = m = \text{直线斜率} = \Delta y / \Delta x</math> (1)</p> <p>那我们就可通过计算x方向上的增量<math>\Delta x</math></p>	<p>(是否引证：否)</p> <p>1. phics/class/course/2-1-1-a.htm 感觉挺麻烦的自己写的话有好数学公式，但代码是自己的啊! 设<math>(x_1, y_1)</math>和<math>(x_2, y_2)</math>分别为所求直线的起点和终点坐标，由直线的微分方程得 <math>m = \text{直线的斜率} = (y_2 - y_1) / (x_2 - x_1)</math> 可通过计算由x方向的增量<math>\Delta x</math>引起y的改变来生成直线：<math>y_2 - y_1 = m(x_2 - x_1)</math></p>
10	<p>此处有 181 字相似</p> <p>增量少见：</p> <p><math>y_{i+1} = y_i + \Delta y</math> (4)</p> <p><math>x_{i+1} = x_i + \Delta x</math> (5)</p> <p>我们<math>x_2 - x_1</math>与<math>y_2 - y_1</math>中较大者作为步进方向(假设<math>x_2 - x_1</math>较大)，取该方向上的增量为一个像素单位(<math>\Delta x = 1</math>)，然后利用式(1)计算另一个方向的增量(<math>\Delta y = \Delta x \cdot m = m</math>)。通过递推公式(2)至(5)，把每次计算出的<math>(x_{i+1}, y_{i+1})</math>经取整后送到显示器输出，则得到扫描转换后的直线。</p> <p>之所以取<math>x_2 - x_1</math>和<math>y_2 - y_1</math>中较大者作为步进方向，是考虑沿着线段分布的像素应均匀。</p> <p>三、抖动消除</p> <p>在了解抖动发生的原因之前这之前，我们需要首先了解3D游戏的发展之路以及一些基本概念。</p> <p>“游戏”一词泛指</p>	<p>直线的DDA算法的C实现 ( opengl实现 ) - zhouyelihua - 博客频道 - CSDN.NET - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - (是否引证：否)</p> <p>1. <math>\Delta y</math> (2 - 4) <math>x_{i+1} = x_i + \Delta x = x_i + \Delta y / m</math> (2 - 5) 式(2 - 2)至(2 - 5)是递推的。选定<math>x_2 - x_1</math>和<math>y_2 - y_1</math>中较大者作为步进方向(假设<math>x_2 - x_1</math>较大)，取该方向上的增量为一个像素单位(<math>\Delta x = 1</math>)，然后利用式(2 - 1)计算另一个方向的增量(<math>\Delta y = \Delta x \cdot m = m</math>)。通过递推公式(2 - 2)至(2 - 5)，把每次计算出的<math>(x_{i+1}, y_{i+1})</math>经取整后送到显示器输出，则得到扫描转换后的直线。之所以取<math>x_2 - x_1</math>和<math>y_2 - y_1</math>中较大者作为步进方向，是考虑沿着线段分布的像素应均匀，这在下图中可看出。另外，算法实现中还应注意直线的生成方向，以决定<math>\Delta x</math>及<math>\Delta y</math>是取正还是负。1、已知直线的两端点坐标：(</p> <p>3DXML标准在协同装配中的研究与应用 曹翔 - 《南京航空航天大学硕士论文》 - 2007-01-01 (是否引证：否)</p> <p>1. ( 3-5 ) 选定 <math>x_2 - x_1</math>和 <math>y_2 - y_1</math>中较大者作为步进方向(假设 <math>x_2 - x_1</math>较大)，取该方向上的增量为一个像素单位(<math>\Delta x = 1</math>)，然后利用式(3 - 1)计算另一个方向的增量(<math>\Delta y = \Delta x \cdot m = m</math>)。通过递推公式(3 - 2)至(3 - 5)，把每次计算出的<math>(x_{i+1}, y_{i+1})</math>经取整后送到显示器输出，则得到扫描转换后的直线。之所以取 <math>x_2 - x_1</math>和 <math>y_2 - y_1</math>中较大者作为步进方向，是考虑沿着线段分布的像素应均匀。另外，算法实现中还应注意直线的生成方向，以决定 <math>\Delta x</math> 和 <math>\Delta y</math> 是取正值还是负值。</p>
11	<p>此处有 42 字相似</p> <p>更新其下一个离散时间的状态。这些结果最后可渲染成图形显示、发出声效或者输出至其他设备。</p> <p>游戏引擎这个术语在20世纪90年代中期形成，这与第一人称射击游戏如id software公司的《DOOM》有关。《DOOM》将其软件构架划分为核心软件组件(如三维图形渲染系统、碰撞检测系统和音频系统等)、美术资产、游戏世界、构成玩家游</p>	<p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》 - 2011-06-01 (是否引证：否)</p> <p>1. Jason在《Game Engine Architecture》一书中说到，游戏引擎的概念产生于90年代中期的第一人称射击游戏，如ids. ftware公司生产的《Doom》151。《0001二》的游戏框架由包括三维图形渲染系统、碰撞检测系统、声音系统在内的许多被合理而明确划分的核心</p>
12	<p>此处有 37 字相似</p> <p>年代中期形成，这与第一人称射击游戏如id software公司的《DOOM》有关。《DOOM》将其软件构架划分为核心软件组件(如三维图形渲染系统、碰撞检测系统和音频系统等)、美术资产、游戏世界、构成玩家游戏体验的游戏规则。这样的划分非常有价值，另一个开发商取得了这样的游戏的授权之后，只需要制作新的美术、关卡布局、</p>	<p>基于HTML5与Node.js的游戏引擎的设计与开发 徐佳宾 - 《南京邮电大学硕士论文》 - 2016-11-18 (是否引证：否)</p> <p>1.e 公司独立研发的。《毁灭战士》游戏的组织架构是特别清晰的，其主要部分可以归纳为：核心软件模块(例如游戏图形渲染模块、碰撞检测模块和音频设置模块等)、UI 设计、游戏场景、游戏规则 ( rule of play )、游戏外设控制。这种就架构体系是很有特点的，比如其他游戏开发商取得这类游戏引擎的</p>

13	<p>此处有 72 字相似</p> <p>维图形渲染系统、碰撞检测系统和音频系统等)、美术资产、游戏世界、构成玩家游戏体验的游戏规则。这样的划分非常有价值,另一个</p> <p>开发商取得了这样的游戏的授权之后,只需要制作新的美术、关卡布局、武器、角色、游戏规则等,对引擎软件做出很少的修改,就可以把游戏打造成新产品。</p> <p>现在主流、常见的商业引擎有:Value公司的Source引擎,Epic的Unreal引擎,以及在移动端很常见的跨平台商业</p>	<p>基于HTML5与Node.js的游戏引擎的设计与开发 徐佳宾 - 《南京邮电大学硕士论文》- 2016-11-18 (是否引证:否)</p> <p>1.1 设计、游戏场景、游戏规则 ( rule of play )、游戏外设控制。这种就架构体系是很有特点的,比如其他游戏开发商取得这类游戏引擎的授权,只需要设计好游戏UI、地图布局、武器样式、角色、操作游戏规则等,对引擎软件做出很少的修改,不用去修改引擎核心,就可以把游戏打造成新产品。这一游戏架构的诞生也引发MOD社区的发展。MOD 的含义是指,某一游戏的众多玩家参与的兴趣小组,或着是小型的独</p>

指 标	
疑似剽窃文字表述	
<p>1. 公式表示为:</p> <p><math>f(n) = g(n) + h(n)</math></p> <p>其中, <math>f(n)</math> 是从初始状态经由状态n到目标状态的代价估计, <math>g(n)</math> 是在状态空间中从初始状态到状态n的实际代价, <math>h(n)</math> 是从状态n到目标状态的最佳路径的估计代价。</p> <p>2. 过程为:</p> <p>1. 把起始格添加到开启列表。</p> <p>2. 寻找开启列表中F值最低的格子,这就是当前格。将这个格子加入到关闭列表。</p> <p>3. 对于当前格相邻的每一个格子,如果这一格不可通过或者已经在关闭列表中,那么就略过这一格;如果这一格不在开启列表中,把这一格添加进去开启列表,并且把当前格作为这一格的父节点,记录这一格的F,G,和H的值;如果这一格已经在开启列表中,</p> <p>3. 如果是,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值;如果你把目标格添加进了关闭列表,则说明路径被找到,或者没有找到目标格,但这个时候开启列表已经空了则说明路径不存在</p> <p>4. 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p> <p>2.2 弗洛伊德路径平滑算法</p> <p>5. 之所以取<math>x_2 - x_1</math>和<math>y_2 - y_1</math>中较大者作为步进方向,是考虑沿着线段分布的像素应均匀。</p> <p>6. 年代中期形成,这与第一人称射击游戏如id software公司的《DOOM》有关。</p> <p>7. 开发商取得了这样的游戏的授权之后,只需要制作新的美术、关卡布局、武器、角色、游戏规则等,对引擎软件做出很少的修改,就可以把游戏打造成新产品。</p>	

脚注和尾注	
1.	[1] 穆俊. 计算机游戏设计原理以及游戏引擎的设计思想[J]. 硅谷,2014,(03):49+98..
2.	[2] 袁俊杰,徐小良. 基于漏斗的实时VBR视频最短路径平滑算法[J]. 计算机系统应用,2010,(07):42-46..

2. 50060943843028650谭鹏_第2部分	总字数: 4363
-----------------------------	-----------

相似文献列表	文字复制比: 10.5%(457)	疑似剽窃观点: (0)
--------	-------------------	-------------

1	基于BREW的手机游戏的研究和实现	6.6% ( 287 )
	孙亭南(导师: 邵时;潘荫荣) - 《华东师范大学硕士论文》- 2006-10-01	是否引证: 否
2	基于VR技术的学生群体紧急疏散模拟研究	6.3% ( 274 )
	江湉湉(导师: 张际平) - 《华东师范大学硕士论文》- 2010-09-01	是否引证: 否
3	A*算法寻找路径	5.9% ( 257 )
	潘海波; - 《黑龙江科技信息》- 2009-06-25	是否引证: 否
4	基于XNA的游戏设计与实现	5.6% ( 244 )
	黄进(导师: 宋成;马迪芳) - 《北京交通大学硕士论文》- 2011-06-01	是否引证: 否
5	深入浅出A*Star算法 - 豆丁网	4.7% ( 206 )
	- 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》- 2015	是否引证: 否
6	第三部分 搜索原理&#40;Part1,2,3,final&#41; - 豆丁网	4.4% ( 194 )
	- 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》- 2016	是否引证: 否
7		4.3% ( 187 )



	A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - 2012	是否引证：否
8	A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - 2011	4.3% ( 187 ) 是否引证：否
9	AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - 2016	4.3% ( 187 ) 是否引证：否
10	A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015	4.3% ( 187 ) 是否引证：否
11	3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012	3.9% ( 169 ) 是否引证：否
12	蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012	3.9% ( 169 ) 是否引证：否
13	群体动画中自主智能体的行为控制 杨延(导师：张建中) - 《电子科技大学硕士论文》 - 2009-06-01	3.7% ( 163 ) 是否引证：否
14	人工智能寻路算法在电子游戏中的研究和应用 詹海波(导师：沈刚) - 《华中科技大学硕士论文》 - 2006-10-01	3.6% ( 159 ) 是否引证：否
15	人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012	3.6% ( 159 ) 是否引证：否
16	个人信用信息法律问题研究 李佳成(导师：蓝寿荣) - 《南昌大学硕士论文》 - 2016-05-28	3.6% ( 156 ) 是否引证：否
17	情感管理视野下高校学生公寓管理研究 沈东益(导师：宋艳) - 《吉林大学硕士论文》 - 2014-04-01	1.9% ( 85 ) 是否引证：否
18	紫花苜蓿耐盐材料的鉴定及胚状体耐盐性诱变研究 张学云(导师：袁庆华) - 《中国农业科学院硕士论文》 - 2013-05-01	1.9% ( 84 ) 是否引证：否
19	基于DSP数字EFO烧球系统的研究与设计 梁付根(导师：赖华) - 《昆明理工大学硕士论文》 - 2016-03-01	1.1% ( 49 ) 是否引证：否
20	蒙古族婚礼服饰研究 贺俊杰(导师：苏和平) - 《中央民族大学硕士论文》 - 2016-06-15	1.1% ( 47 ) 是否引证：否

原文内容		相似内容来源
1	<p>此处有 169 字相似</p> <p>控制摄像机沿着平滑的路径，以平滑的速度运动。 对于平滑的路径，我们使用A*寻路找到这样的路径。采用以下步骤：</p> <p>1. 把 起始格添加到开启列表。</p> <p>2. 寻找开启列表中F值最低的格子，这就是当前格。将这个格子加入到关闭列表。</p> <p>3. 对于当前格相邻的每一个格子，如果这一格不可通过或者已经在关闭列表中，那么就略过这一格；如果这一格不在开启列表中，把这一格添加进去开启列表，并且把当前格作为这一格的父节点，记录这一格的F,G,和H的值；如果这一格已经在开启列表中，</p> <p>则检查这一格的G值是否更低，如果是，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值；如果你把目标格添加进了关</p>	<p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 ( 是否引证：否 )</p> <p>1.方法被称为启发式的原因。2.A*方法过程 (1)把起始格添加到开启列表。(2)重复如下的工作: a)寻找开启列表中F值最低的格子。称它为当前格。b)把它切换到关闭列表。c)对相邻的8格中的每一个。如果它不可通过或者已经在关闭列表中，略过它。反之，①如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F，G，和H值;②如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，</p>
		<p>A*算法寻找路径 潘海波; - 《黑龙江科技信息》 - 2009-06-25 ( 是否引证：否 )</p> <p>1.。另一方面,如果新的G值更低,那就把相邻方格的父节点改为目前选中的方格。最后,重新计算F和G的值。 A*方法总结:(1)把起始格添加到开启列表。(2)重复如下的工作:a.寻找开启列表中F值最低的格子。称它为当前格。b.把它切换到关闭列表。c.对相邻的8格中的每一个:如果它不可通过或者已经在关闭列表中,略过它。反之如下。如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一</p>

	<p>基于VR技术的学生群体紧急疏散模拟研究 江湑湑 - 《华东师范大学硕士论文》 - 2010-09-01 (是否引证：否)</p> <p>1. 341把起始格添加到开启列表。·—2重复如下的工作：  a)寻找开启列表中F值最低的格子，称为当前格。_ b)把它切换到关闭列表。_ c)对相邻的8格中的每一个：·如果它不可通过或者已经在关闭列表中，略过它。反之如下。·如果它不在开启列表中，把它添加进去。把当前格作设这一格的父节点。记录这一格的F、G和H值，·如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点</p> <p>A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - (是否引证：否)</p> <p>1.把起始格添加到开启列表。2，重复如下的工作： a) 寻找开启列表中F值最低的格子。我们称它为当前格。  b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个？  * 如果它不可通过或者已经在关闭列表中，略过它。反之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。  * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p> <p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - (是否引证：否)</p> <p>1.把起始格添加到开启列表。2，重复如下的工作： a) 寻找开启列表中F值最低的格子。我们称它为当前格。  b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个？  * 如果它不可通过或者已经在关闭列表中，略过它。反之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。  * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p> <p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - (是否引证：否)</p> <p>1.把起始格添加到开启列表。2，重复如下的工作： a) 寻找开启列表中F值最低的格子。我们称它为当前格。  b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个？  * 如果它不可通过或者已经在关闭列表中，略过它。反之如下。 * 如果它不在开启列表中，把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。  * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一</p> <p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》 - 2011-06-01 (是否引证：否)</p> <p>1.作: a)寻找开启列表中F值最低的格子。我们称它为当前格。 b)把它一切切换到关闭列表。 C)对相邻的8格中的</p>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

		<p>每一个格子进行检测:如果它不可通过或者已经在 关闭列表中, 略过它。如果它不在开启列表中, 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F, G, 和 H 值。如果它已经在开启列表中, 用 G 值为参考检查新的路径是否更好。更低的 G 值 意味着更好的路径。如果是这样, 就把这一格的父节</p>
		<p>深入浅出A*Star算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015-7-24 0:28:46 ( 是否引证 : 否 )</p>
		<p>1.始格添加到开启列表。 ,2、 重复如下的工作,a) 寻找开启列表中F值最低的格子。我们称它为当前格。 ,b) 把它切换到关闭列表。 ,c) 对相邻的8格中的每一个,如果它不可通过或者已经在关闭列表中,略过它。反之如下。 ,如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 ,如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一</p>
		<p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-16 12:07:06 ( 是否引证 : 否 )</p>
		<p>1.作 : a) 寻找开启列表中 F 值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的 8 格中的每一个 ? * 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 * 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父 节点。记录这一格的 F,G,和 H 值。 * 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。 更低的 G 值意味着更好的路径。 如果是这样 , 就把这一格的父节点改成当前格 ,</p>
		<p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-19 23:58:36 ( 是否引证 : 否 )</p>
		<p>1.的工作 : a) 寻找开启列表中 F 值最低的格子。我们称它为当前格。 b) 把它切换到 关闭列表。 c) 对相邻的 8 格中的每一个 ? 如果它不可通过或者已经在关闭列表中 , 略过它。反之如下。 如果它不在开启列表中 , 把它添加进去。把当前格作为这一格的父节点。记录这一格的 F,G,和 H 值。 如果它已经在开启列表中 , 用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样 , 就把这一格的父节点改成当前格 , 并且重</p>
		<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证 : 否 )</p>
		<p>1.把起始格添加到开启列表。 2. 重复如下的工作,a) 寻找开启列表中F值最低的格子。我们称它为当前格。 b) 把它切换到关闭列表。 c) 对相邻的8格中的每一个,1. 如果它不可通过或者已经在关闭列表中,略过它。反之如下。 2. 如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 3. 如果它已经在开启列表中,用G值为参考检查新的路径是否</p>

		<p>更好。更低的G值意味着更好的路径。如果是这样,就把这</p> <p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 (是否引证:否)</p> <p>1.列表。(c)对相邻的8格中的每一格进行如下操作: * 如果它不可通过或者已经在关闭列表中,略过它。反之如下。 * 如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点</p> <p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学硕士论文》 - 2009-06-01 (是否引证:否)</p> <p>1. c) 对相邻的8格中的每一个格子进行如下判断: * 如果它不可通过或者已经在关闭列表中,略过它。反之如下。 * 如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点</p> <p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源(<a href="http://www.docin.com">http://www.docin.com</a>)》 - 2012-11-7 7:01:49 (是否引证:否)</p> <p>1.列表中值最低的格子。我们称它为当前格。(b)把它切换到关闭列表。(c)对相邻的8格中的每一格进行如下操作: * 如果它不可通过或者已经在关闭列表中,略过它。反之如下。 * 如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和H值。 * 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一</p> <p>第三部分 搜索原理#40;Part1,2,3,final#41;- 豆丁网 - 《互联网文档资源(<a href="http://www.docin.com">http://www.docin.com</a>)》 - 2016-10-28 11:00:02 (是否引证:否)</p> <p>1.。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。机器人与信息自动化研究所自动化与智能科学系,如果它不可通过或者已经在关闭列表中,略过它。反之如下。如果它不在开启列表中,把它添加进去。把当前格作为这一格的父节点。记录这一格的F,G,和HH值。如果它已经在开启列表中,用GG值为参考检查新的路径是否更好。更低的GG值意味着更好的路径。如果是这样,就</p>
2	<p>此处有 90 字相似</p> <p>,并且把当前格作为这一格的父节点,记录这一格的F,G,和H的值;如果这一格已经在开启列表中,则检查这一格的G值是否更低,</p> <p>如果是,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值;如果你把目标格添加进了关闭列表,则说明路径被找到,或者没有找到目标格,但这个</p>	<p>A*算法寻找路径 潘海波;-《黑龙江科技信息》- 2009-06-25 (是否引证:否)</p> <p>1.节点。记录这一格的F,G,和H值。如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d.停止:把目标格添加进了开启列表</p>



<p>时候开启列表已经空了则说明路径不存在</p> <p>则结束这一步执行4。遍历完所有邻格后，执行2。</p> <p>4. 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这</p>	<p>,这时候路径被找到</p> <p>2.前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d.停止:把目标格添加进了开启列表,这时候路径被找到。没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。责任编辑:王青翠A*算法寻找路径@潘</p> <p>基于VR技术的学生群体紧急疏散模拟研究 江湑湑 - 《华东师范大学硕士论文》 - 2010-09-01 (是否引证:否)</p> <p>1.如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。d)停止,当:把目标格添加进了开启列表,这时候路径被找到,或者。没有找到目标格,开启列表已经空了。这时候,路径不存在。3保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是所求的路径</p> <p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》 - 2011-06-01 (是否引证:否)</p> <p>1.经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。</p> <p>2.表按F值排序,改变之后你可能需要重新对开启列表排序。d)停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。3)保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这</p> <p>深入浅出A*Star算法 - 豆丁网 - 《互联网文档资源 (http://www.docin.com)》 - 2015-7-24 0:28:46 (是否引证:否)</p> <p>1.记录这一格的F,G,和H值。如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d)停止,当你把目标格添加进了开启列表,这时候</p> <p>2.并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。d)停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。3、保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。题外话,</p> <p>A*寻路初探 - 自由天地 - 《网络 (http://blog.donews.c)》 -</p>
-----------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<div data-bbox="837 38 1543 707"> <p>(是否引证：否)</p> <p>1.记录这一格的F,G,和H值。* 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。</b> d) 停止，当你 * 把目标格添加进了开启列表，这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 * 把<b>目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了</b>。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题一</p> </div> <div data-bbox="837 707 1543 1402"> <p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - (是否引证：否)</p> <p>1.记录这一格的F,G,和H值。* 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。</b> d) 停止，当你 * 把目标格添加进了关闭列表，这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 * 把<b>目标格添加进了关闭列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了</b>。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。(注解:在这篇</p> </div> <div data-bbox="837 1402 1543 2096"> <p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - (是否引证：否)</p> <p>1.记录这一格的F,G,和H值。* 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。</b> d) 停止，当你 * 把目标格添加进了开启列表，这</p> <p>2.计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 * 把<b>目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了</b>。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p> </div> <div data-bbox="837 2096 1543 2159"> <p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 (是否引证：否)</p> </div>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<p>1.H值;②如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果保持开启列表按F值排序，改变之后可能需要重新对开启列表排序。</b></p> <p>2.序，改变之后可能需要重新对开启列表排序。 d)停止当把<b>目标格添加进了开启列表，这时候路径被找到。或者没有找到目标格，开启列表已经空了。</b>这时候，路径不存在。(3)保存路径。从目标格开始，沿着每一格的父节点移动直到回到起</p>
	<p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证：否 )</p>
	<p>1.记录这一格的F,G,和H值。 * 如果它已经在开启列表中，用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。</b>（ d ） 停止，当我们 * 把目标格添加进了关闭列表</p> <p>2.这一格的G和F值。如果你保持你的开启列表按F值排序，改变之后你可能需要重新对开启列表排序。（ d ） 停止，当我们 * 把<b>目标格添加进了关闭列表这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。</b>这时候，路径不存在。（ 3 ） 保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是要求的最短路径。 2.4</p>
	<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证：否 )</p>
	<p>1.录这一格的F,G,和H值。 3. 如果它已经在开启列表中,用G值为参考检查新的路径是否更好。更低的G值意味着更好的路径。<b>如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。</b> d) 停止,当你 4. 把目标格添加进了开启列表,</p> <p>2.算这一格的G和F值。如果你保持你的开启列表按F值排序,改变之后你可能需要重新对开启列表排序。 d) 停止,当你 4. 把<b>目标格添加进了开启列表,这时候路径被找到,或者 5. 没有找到目标格,开启列表已经空了。</b>这时候,路径不存在。 3. 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题</p>
	<p>第三部分 搜索原理&amp;#40;Part1,2,3.final&amp;#41;- 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2016-10-28 11:00:02 ( 是否引证：否 )</p>
	<p>1.一格的F,G,和HH值。 ,如果它已经在开启列表中,用GG值为参考检查新的路径是否更好。更低的GG值意味着更好的路径。<b>如果是这样,就把这一格的父节点改成当</b></p>

	<p>前格,并且重新计算这一格的GG和和FF值。如果你保持你的开启列表按FF值排序,改变之后你可能需要重新对开启列表排序。 ,停止,当你把目标格添加进</p> <p>2.重新计算这一格的GG和和FF值。如果你保持你的开启列表按FF值排序,改变之后你可能需要重新对开启列表排序。 ,停止,当你把目标格添加进了开启列表,这时候路径被找到,或者 没有找到目标格,开启列表已经空了。这时候,路径不存在。 机器人与信息自动化研究所 自动化与智能科学系 ,计算机科学 ,优化调度 ,运输问题 ,组合优化 ,</p>
	<p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p>
	<p>1.* 如果它已经在开启列表中,用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的 G 和 F 值。如果你保持你的开启列表按 F 值排序,改变之后你可能需要重新对开启列表排序。</p> <p>2.新对开启列表排序。( d ) 停止,当我们 * 把目标格添加进了关闭列表,这时候路径被找到,或者 * 没有找到目标格,开启列表已经空了。这时候,路径不存在。</p> <p>( 3 ) 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就</p>
	<p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学硕士论文》 - 2009-06-01 ( 是否引证 : 否 )</p>
	<p>1. * 如果它已经在开启列表中,用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的 G 和 F 值。如果保持你的开启列表按 F 值排序,改变之后可能需要重新对开启列表排序。</p> <p>2.列表按 F 值排序,改变之后可能需要重新对开启列表排序。 d) 停止,当目标格添加进了关闭列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。 3 ) 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你</p>
	<p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-16 12:07:06 ( 是否引证 : 否 )</p>
	<p>1.G,和 H 值。 * 如果它已经在开启列表中,用 G 值为参考检查新的路径是否更好。 更低的 G 值意味着更好的路径。 如果是这样,就把这一格的父节点改成当前格,并且重新计算这一格的 G 和 F 值。如果你保持你的开启列表按 F 值排序,改变之后你可能需要重新对开启列表排序。d) 停止,当你 47 北京</p> <p>2.启列表按 F 值排序,改变之后你可能需要重新对开启列表排序。d) 停止,当你 47 北京理工大学工程硕士学位论文 * 把目标格添加进了开启列表,这时候路径被找到,或者 * 没有找到目标格,开启列表已经空了。这时候,路径不存在。 3. 保存路径。从目标格开始,沿着</p>



		<p>每一格的父节点移动直到回到起始格。这就是你的路径。 。 1.4 题</p> <p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》 - 2012-11-19 23:58:36 ( 是否引证：否 )</p> <p>1. F,G,和 H 值。 如果它已经在开启列表中，用 G 值为参考检查新的路径是否更好。更低的 G 值意味着更好的路径。如果是这样，就把这一格的父节点改成当前格，并且重新计算这一格的 G 和 F 值。如果你保持你的开启列表按 F 值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 把目标</p> <p>2.的 G 和 F 值。如果你保持你的开启列表按 F 值排序，改变之后你可能需要重新对开启列表排序。 d) 停止，当你 把目标格添加进了开启列表，这时候路径被找到，或者 没有找到目标格，开启列表已经空了。这时候，路径不存在。保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题一下，</p>
3	<p>此处有 41 字相似</p> <p>，或者没有找到目标格，但这个时候开启列表已经空了则说明路径不存在则结束这一步执行4。遍历完所有邻格后，执行2。</p> <p>4.</p> <p>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p> <p>该过程的位代码为：</p> <p>开启列表 ← 起始格</p> <p>关闭列表 ← {}</p> <p>repeat</p> <p>当前格 ← min(F(开启列表))</p>	<p>A*寻路初探 - 自由天地 - - 《网络 ( <a href="http://blog.donews.c">http://blog.donews.c</a> ) 》 - ( 是否引证：否 )</p> <p>1. * 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题一下，见谅，值得一提的是，当你在网上或者相关论坛看到关于A*的不同的探讨，你有时会看到一些被当作A*算法的</p> <p>A*寻路算法初探 - 思月行云 - CSDN博客 - 《网络 ( <a href="http://blog.csdn.net">http://blog.csdn.net</a> ) 》 - ( 是否引证：否 )</p> <p>1. * 把目标格添加进了关闭列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。 (注解:在这篇文章的较早版本中，建议的做法是当目标格 ( 或节点 ) 被加入到开启列表，而不是关闭列表的时候停止寻路。这么做会</p> <p>AStar寻径算法小结_leiii - 《网络 ( <a href="http://blog.sina.com">http://blog.sina.com</a> ) 》 - ( 是否引证：否 )</p> <p>1. * 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。 3.保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</p> <p>基于BREW的手机游戏的研究和实现 孙亭南 - 《华东师范大学硕士论文》 - 2006-10-01 ( 是否引证：否 )</p> <p>1.找到目标格，开启列表已经空了。这时候，路径不存在。(3)保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是移动的路径。 3.A*实现 (1)维护开启列表: 这是</p> <p>群体动画中自主智能体的行为控制 杨延 - 《电子科技大学</p>

	<div data-bbox="837 40 1543 73" data-label="Text"> <p>硕士论文》- 2009-06-01 ( 是否引证：否 )</p> </div> <div data-bbox="837 73 1543 347" data-label="Text"> <p>1.进了关闭列表，这时候路径被找到，或者没有找到目标格，开启列表已经空了。这时候，路径不存在。 3 ) <b>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 在介绍完 A*算法的实现原理后，下面具体给出 A*算法的实现代码。 60</p> </div> <div data-bbox="837 347 1543 409" data-label="Text"> <p>A*算法寻找路径 潘海波; - 《黑龙江科技信息》- 2009-06-25 ( 是否引证：否 )</p> </div> <div data-bbox="837 409 1543 723" data-label="Text"> <p>1.排序。d.停止:把目标格添加进了开启列表,这时候路径被找到。没有找到目标格,开启列表已经空了。这时候,路径不存在。(3)<b>保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 责任编辑:王青翠A*算法寻找路径@潘海波\$黑龙江旅游职业技术学院!黑龙江哈尔滨150086介绍的就是实现如何用A*算法在</p> </div> <div data-bbox="837 723 1543 786" data-label="Text"> <p>基于VR技术的学生群体紧急疏散模拟研究 江湑湑 - 《华东师范大学硕士论文》- 2010-09-01 ( 是否引证：否 )</p> </div> <div data-bbox="837 786 1543 1059" data-label="Text"> <p>1.了开启列表，这时候路径被找到，或者。没有找到目标格，开启列表已经空了。这时候，路径不存在。3<b>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是所求的路径。</b> A*算法是静态的寻路算法，所谓静态的指的是在环境不变(静态环境)的情况下所进行的算法。</p> </div> <div data-bbox="837 1059 1543 1122" data-label="Text"> <p>基于XNA的游戏设计与实现 黄进 - 《北京交通大学硕士论文》- 2011-06-01 ( 是否引证：否 )</p> </div> <div data-bbox="837 1122 1543 1357" data-label="Text"> <p>1.找到，或者没有找到目标格，开启列表已经空了。这时候，路径不存在。3)<b>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 。3.蚁群算法 蚁群算法(A, It eoroll)optil二 izatio , 1.ACo)是一种最新</p> </div> <div data-bbox="837 1357 1543 1447" data-label="Text"> <p>3D游戏引擎设计及其关键技术-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》- 2012-11-16 12:07:06 ( 是否引证：否 )</p> </div> <div data-bbox="837 1447 1543 1760" data-label="Text"> <p>1.* 把目标格添加进了开启列表，这时候路径被找到，或者 * 没有找到目标格，开启列表已经空了。这时候，路径不存在。3. <b>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 1.4 题外话 值得一提的是，当你在网上或者相关论坛看到关于 A*的不同的探讨，你有时会看到一些被当作 A*算法的代</p> </div> <div data-bbox="837 1760 1543 1850" data-label="Text"> <p>蚁群算法-百度文库 - 《互联网文档资源 ( <a href="http://wenku.baidu.c">http://wenku.baidu.c</a> ) 》- 2012-11-19 23:58:36 ( 是否引证：否 )</p> </div> <div data-bbox="837 1850 1543 2159" data-label="Text"> <p>1. 停止，当你 把目标格添加进了开启列表，这时候路径被找到，或者 没有找到目标格，开启列表已经空了。这时候，路径不存在。 <b>保存路径。从目标格开始，沿着每一格的父节点移动直到回到起始格。这就是你的路径。</b> 。题外话 离题一下，见谅，值得一提的是，当你在网上或者相关论坛看到关于 A*的不同的 探讨，你有时会看到一些被当作 A*</p> </div>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

		<p>A寻路算法.doc - 《互联网文档资源 ( <a href="http://max.book118.c">http://max.book118.c</a> ) 》 - 2015-12-6 4:28:56 ( 是否引证 : 否 )</p> <p>1. 把目标格添加进了开启列表,这时候路径被找到,或者 5. 没有找到目标格,开启列表已经空了。这时候,路径不存在。 3. 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 题外话 离题一下,见谅,值得一提的是,当你在网上或者相关论坛看到关于A*的不同的探讨,你有时会看到一些被当作A*算法的</p> <p>深入浅出A*Star算法 - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2015-7-24 0:28:46 ( 是否引证 : 否 )</p> <p>1.停止,当你把目标格添加进了开启列表,这时候路径被找到,或者没有找到目标格,开启列表已经空了。这时候,路径不存在。 ,3、 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 ,题外话 ,离题一下,见谅,值得一提的是,当你在网上或者相关论坛看到关于A*的不同的探讨,你有时会看到一些被当作A*算</p> <p>第三部分 搜索原理Part1,2,3,final - 豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2016-10-28 11:00:02 ( 是否引证 : 否 )</p> <p>1.作,寻找开启列表中FF值最低的格子。 ,把它切换到关闭列表。 ,对相邻的88格中的每一个进行检查,检查过程见下页。 ,保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是你的路径。 机器人与信息自动化研究所 自动化与智能科学系 ,如果它不可通过或者已经在关闭列表中,略过它。反之如下。 ,如果它不在开</p> <p>人工智能寻路算法在电子游戏中的研究和应用 詹海波 - 《华中科技大学硕士论文》 - 2006-10-01 ( 是否引证 : 否 )</p> <p>1. * 没有找到目标格,开启列表已经空了。这时候,路径不存在。 ( 3 ) 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是要求的最短路径。 2.4 本章小结 本章主要介绍了电子游戏中人工智能寻路算</p> <p>人工智能寻路算法在电子游戏中的研究和应用 - docin.com豆丁网 - 《互联网文档资源 ( <a href="http://www.docin.com">http://www.docin.com</a> ) 》 - 2012-11-7 7:01:49 ( 是否引证 : 否 )</p> <p>1. * 把目标格添加进了关闭列表这时候路径被找到,或者 * 没有找到目标格,开启列表已经空了。这时候,路径不存在。 ( 3 ) 保存路径。从目标格开始,沿着每一格的父节点移动直到回到起始格。这就是要求的最短路径。 2.4 本章主要介绍了电子游戏中人工智能寻路算法涉及的人工智能、图的遍历算法,最短路径搜索等关键性技术</p> <p>个人信息法律问题研究 李佳成 - 《南昌大学硕士论文》 - 2016-05-28 ( 是否引证 : 否 )</p> <p>1.究方法,还使我明白了许多待人接物与为人处事的道理。本论文从选题到完成,每一步都是在导师的指导下</p>
4	<p>此处有 157 字相似</p> <p>提供了很大的帮助,在过程中每周五都会在望江实验室进行交流、了解进度、对不清楚的部分进行指导、说明</p>	

<p>。这篇文章的成形，离不开老师的心血。在此，谨向导师表示崇高的敬意和衷心的感谢!</p> <p>通过这篇论文的撰写，使我能够等系统、全面的学习有关游戏开发的最新的、最先进的前沿理论知识，这对于我今后的工作和和我以后自身的发展，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的有些观点和归纳和阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。</p>	<p>完成的，倾注了导师大量的心血。在此谨向导师表示崇高的敬意和衷心的感谢！通过论文的撰写，使我能够更系统、全面地学习有关法律新型的、先进的前沿理论知识，并得以借鉴众多专家学者的宝贵经验，这对于我今后的工作，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的</p> <p>2.通过论文的撰写，使我能够更系统、全面地学习有关法律新型的、先进的前沿理论知识，并得以借鉴众多专家学者的宝贵经验，这对于我今后的工作，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的有些观点的归纳和阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。李佳成2016年5月参考文献31参考文献（一）著作类[1]白云：《个人信息信息法律保</p>
	<p>情感管理视野下高校学生公寓管理研究 沈东益 -《吉林大学硕士论文》- 2014-04-01 (是否引证：否)</p> <p>1.取更多的营养，从而能够为自己进一步地加油充电。通过论文的撰写，使我能够更系统、全面地学习有关公共事业管理方面知识，这对于我今后的工作有很大的帮助，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的有些观点归纳和阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。36</p>
	<p>紫花苜蓿耐盐材料的鉴定及胚状体耐盐性诱变研究 张学云 -《中国农业科学院硕士论文》- 2013-05-01 (是否引证：否)</p> <p>1.通过论文的撰写，使我能够更系统、全面地学习有关草业科学先进的前沿理论知识，并得以借鉴众多专家学者的宝贵经验，这对于我今后的学习和工作，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的有些观点和归纳阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。最后，衷心地感谢评委和答辩委员会老师，您辛苦了！张学云2013年5月于北京57中国农业科</p>
	<p>基于DSP数字EFO烧球系统的研究与设计 梁付根 -《昆明理工大学硕士论文》- 2016-03-01 (是否引证：否)</p> <p>1.学在软件调试的过程中给予了我的帮助。正是这些研究生同学在学习的过程中相互讨论与鼓励才使我的学习知识面得到拓展。由于本人理论水平比较有限，论文中的有些观点及对设背的归纳和阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。在此，我对您百忙之中对我论文的查阅与指正表示深深地感谢。65昆明理工大学硕士±学位论文</p>
	<p>蒙古族婚礼服饰研究 贺俊杰 -《中央民族大学硕士论文》- 2016-06-15 (是否引证：否)</p> <p>1.锦。更要感谢生我养我的父母，他们给予了我无私的爱，为我的成长付出了许许多多，祝愿他们健康长寿！由于本人理论水平比较有限，论文中的有些观点的归纳和阐述难免有疏漏和不足的地方，欢迎各位老师和专家们提出批评与指正。66攻读学位期间发表的学术论文目录[1] 贺俊杰:八白族服饰的演变及其图案文化浅</p>





--	--	--

指 标

疑似剽窃文字表述

1. 起始格添加到开启列表。
2. 寻找开启列表中F值最低的格子，这就是当前格。将这个格子加入到关闭列表。
3. 对于当前格相邻的每一个格子，如果这一格不可通过或者已经在关闭列表中，那么就略过这一格；如果这一格不在开启列表中，把这一格添加进去开启列表，并且把当前格作为这一格的父节点，记录这一格的F,G,和H的值；如果这一格已经在开启列表中，
2. 如果是，就把这一格的父节点改成当前格，并且重新计算这一格的G和F值；如果你把目标格添加进了关闭列表，则说明路径被找到，或者没有找到目标格，但这个时候开启列表已经空了则说明路径不存在
3. 老师的心血。在此，谨向导师表示崇高的敬意和衷心的感谢！
- 通过这篇论文的撰写，使我能够等系统、全面的学习有关游戏开发的最新的、最先进的前沿理论知识，这对于我今后的工作和和我以后自身的发展，无疑是不可多得的宝贵财富。由于本人理论水平比较有限，论文中的有些观点和归纳和阐述难免有疏漏和不足的地方，欢迎老师和专家们指正。

说明：1.仅可用于检测期刊编辑部来稿，不得用于其他用途。

- 2.总文字复制比：被检测论文总重合字数在总字数中所占的比例。
- 3.去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例。
- 4.去除本人已发表文献复制比：去除作者本人已发表文献后，计算出来的重合字数在总字数中所占的比例。
- 5.指标是由系统根据《学术期刊论文不端行为的界定标准》自动生成的。
- 6.红色文字表示文字复制部分;绿色文字表示引用部分。
- 7.本报告单仅对您所选择比对资源范围内检测结果负责。
- 8.Email：[amlc@cnki.net](mailto:amlc@cnki.net)       <http://e.weibo.com/u/3194559873>       [http://t.qq.com/CNKI\\_kycx](http://t.qq.com/CNKI_kycx)