要求

使用JavaScript的程序实现一个简单的游戏，包括三个阶段，setup，play和end。在play阶段游戏按回合（rounds）进行。游戏在一个由墙壁包围的10×10的网格上进行，包含我们由用户控制的hero，和一群邪恶的通过计算机（也就是你的程序）控制的killer robot。Killer robot追捕我们的hero，而hero寻找power-up暂时让hero可以摧毁killer robot。用户与你的程序是本场游戏的两名选手。

游戏始终由setup阶段开始。在该阶段，用户可放置三种不同类型对象在网格的单元格中：

1.通过点击一个单元格并输入1到9之间的一个数字，一个power-up 将被放置在此单元格中，数字表示power-up的价值;

2.通过单击单元格并键入字母“O”，一个障碍物被放置在此单元格;

3.通过单击单元格并键入字母“H”，hero被放置在此单元格。

4.通过点击一个小区并键入字母“K”，在一个killer robot被放置在此单元格。

Power-up，障碍和killer-robot的数量没有限制，但是只能有一个hero。没有单元格可以放置超过一个对象，一旦一个对象被放置在单元格内将不能被改变。如果用户试图改变放置在单元格中的对象，则会显示错误。如果用户键入的字符不是间1至9和“o”，“H”和“k”，也会提示输入错误。

用户必须具有结束setup阶段的方法，例如通过一个按钮。如果用户没有放置hero就试图结束比赛的setup阶段，那么应该显示一条错误消息，并且将用户保留在setup阶段。或者，游戏继续进入play阶段。

在play阶段开始和期间，再次显示网格和其上放置的对象，以及附加信息：至今的回合数（rounds）和用户得分（score）。最初回合数和用户的得分都为零。另外，用户必须可以随时结束play阶段，例如可能通过一个按钮。

而在play阶段，游戏按回合（rounds）进行，每轮由用户开始随后才是计算机。

轮到他/她时，用户可以尝试通过输入字母在网格中水平或垂直移动hero：

“a”hero向左移动一格，

“d”hero向右移动一格，

“w”hero向上移动一格，

“x”hero向下移动一格。

如果用户输入任何其它字符，那么应显示一个错误消息。如果所尝试的移动将导致hero到网格以外或遇到障碍物，那么一个错误消息应当被显示，移动失败，主人公不动，并且将轮到killer robot移动。若移动成功，hero的位置将更改。如果hero停留在放置了power-up的格子，那么power-up将被删去，power-up的值n将被添加到用户的分数里（score），hero将变成为一个superhero并持续n轮时间，本轮作为计数第一轮。如果hero停在killer robot占用的格子里并且他目前是一个superhero，那么killer robot将被摧毁，从网格中消失，用户的得分（score）增加一百点。如果hero在不是superhero时停在killer robot的格子里，那么英雄死掉，游戏结束，比赛进行到end阶段。英雄改​​变了位置（比如移动一格），或者收集了power-up或死了，那么就轮到killer robot。

轮到计算机时，你的程序尝试以某种顺序移动每一个killer robot。与hero不同，杀手机器人不仅能够在水平和垂直方向移动也可以走斜线。每个killer robot一轮最多移动一格。如hero在killer robot紧挨的单元格内，然后killer robot必须移动到hero所在的单元格，如果hero现在是superhero，那么killer robot被毁灭，从网格中消失，用户得分增加100分;如果hero当前不是superhero，hero死去，次回合结束，回合数增加一，游戏到end阶段。如果hero不在killer robot紧挨的单元格中，但其中一个或多个单元格中包含power-up，那么killer robot必须移动到这些单元格其中之一，并将该单元格中的power-up删除（从而杀手机器人剥夺hero收集power-up的机会）。如果没有紧挨的单元格包含hero或者power-up，那么killer robot可以移动到任意的周边单元格，此举不会使它超出网格或到有障碍物和另一个killer robot的单元格。如果Killer robot可以移动，不准站着不动。但是，如果一个killer robot不能动，那么电脑应该直接操控下一个killer robot。一旦已尝试向移动过每个杀手机器人，那么就轮到用户，并且回合数加一。

在每个回合或回合内的每轮结束，程序显示的统计信息必须更新。

如果满足下列任意条件那么play阶段结束：

1.用户结束play阶段（通过按按钮）;

2.hero死亡;

3.网格内没有killer robot了;

4.网格内没有power-up了;

5.hero和任何一个killer robot都无法移动了。

一旦play阶段结束，游戏进入end阶段。到end阶段程序判定游戏的结果。如果没有killer robot或没有power-up留在网格，并且hero是活的，那么用户胜利; 如果hero死了，那么计算机取胜;否则，结果是一个平局（draw）。该程序应显示一条消息，表明游戏结果，然后停止。在end阶段任何用户输入或动作程序都没有反应。

1.可能在setup阶段用户不在网格上放置任何power-up。在进入play阶段时程序就应该意识到，并立即进入end阶段，并宣布用户赢得比赛。

2.也有可能在setup阶段用户不在网格上放置任何killer robot。在进入play阶段时程序就应该意识到，并立即进入end阶段，并宣布用户赢得比赛。

3.你应该仔细分析hero和killer robot无法移动的情况，以正确结束play阶段。

4.理想情况下你的程序会以增加杀死hero几率的方式移动killer robot。这可能是由每个killer robot每次移动都试图减小和hero的距离来实现。但你也可以实现一个策略，通过killer robot'围剿'hero，以增加他们杀hero的机会。很明显，当hero是一个superhero，killer robot应尽量保持和hero的安全距离。

5.理想的情况下的程序将允许网格的尺寸容易地改变（由系统的维护者），独立地对每个dimension。

6.JavaScript引擎在不同的浏览器下不同。你应该确保你的系统在所有常用的浏览器（例如，谷歌Chrome，Mozilla Firefox浏览器，微软Internet Explorer 9或更高版本）和所有常用的平台（例如，Linux的衍生工具和Microsoft Windows）上都能工作。

7.你的JavaScript程序应该只取决于你自己的代码。 JavaScript libraries／frameworks不能使用。

8.您的代码应加入适当评论。这包括指出你的代码已被开发出的上线来源或教科书和参考这些源的帮助。

评定

该JavaScript程序可通过要求的URL访问并且不产生错误信息：8

解决JavaScript编程任务的正确性和质量（92）：

1.interface设计的质量：7

2.setup阶段实施的正确性和质量：14

3.play阶段的实施正确性和质量：44

4.检测游戏的结束和end阶段的正确性和质量：15

5.良好的格式，注释和代码质量：12