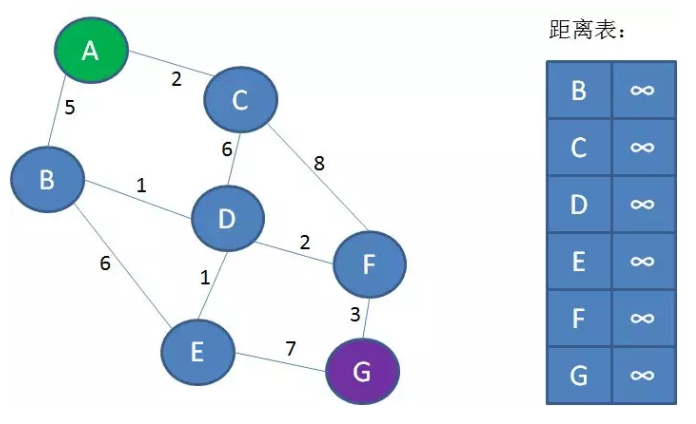
# 图的最短路径问题

详情：https://www.cnblogs.com/xiaozhongfeixiang/p/11725665.html

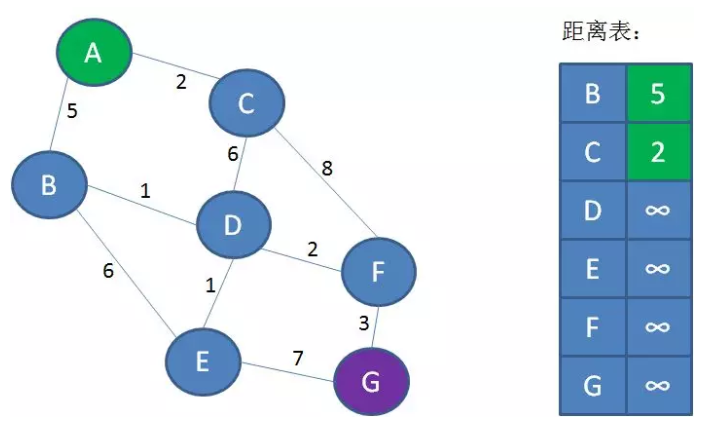
究竟什么是迪杰斯特拉算法？它是如何寻找图中顶点的最短路径呢？

dijstra(迪杰斯特拉)算法步骤：

1、创建距离表。key是顶点名称，value是从起点A到对应顶点的已知最短距离。A到其它顶点距离初始为无限大。

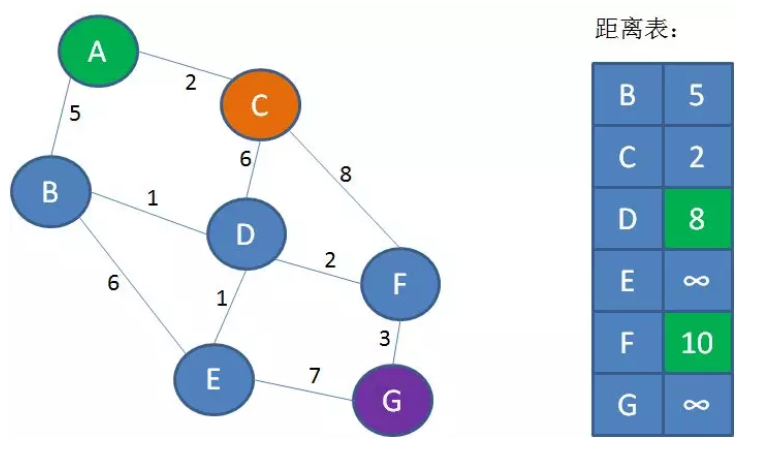


2、遍历起点A，找到起点A的邻接点B和C，将BC到A的距离更新到距离表中。



3、从距离表中找到从A出发距离最短的点，即顶点C（A已经遍历过了，不需要考虑）。

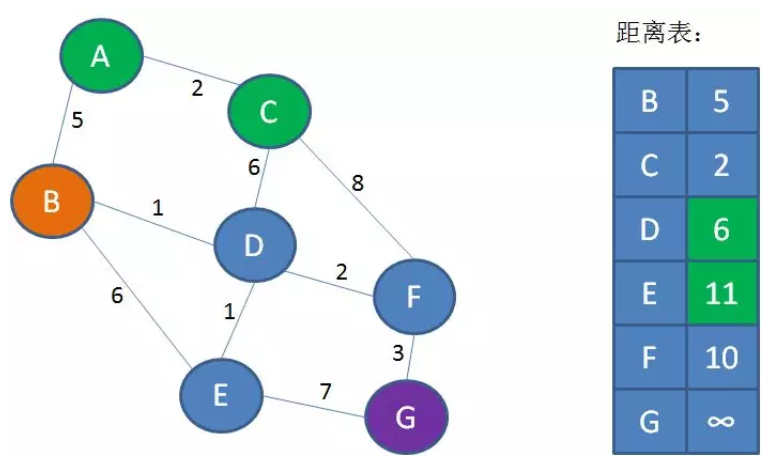
4、遍历顶点C，找到邻接点D和F。从C到D距离是6，所以从A到D距离是2+6=8；从C到F距离是8，所以从A到F距离是2+8=10。把这一信息刷新到表中。



重复步骤3和步骤4的操作。

5、遍历距离表找到从A出发距离最短的点，即顶点B（AC已经遍历过了，不需要考虑）。

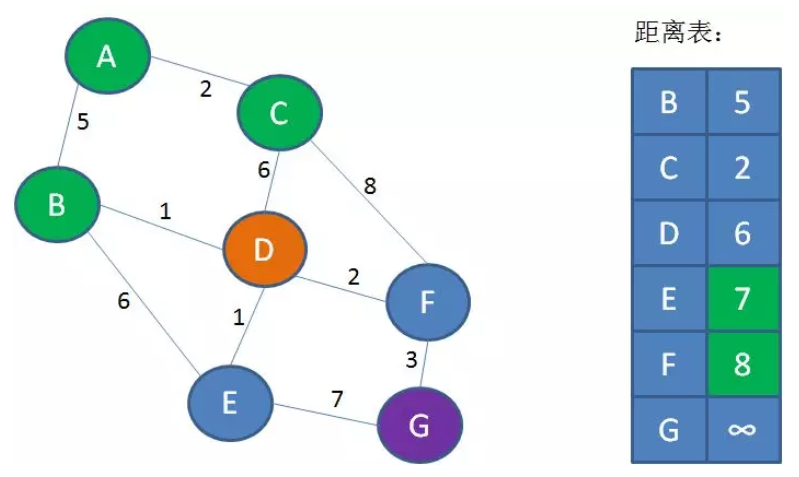
6、遍历顶点B，找到邻接点D和E。从B到D距离是1，所以A到D的距离是5+1=6（小于距离表中的8）；从B到E的距离是6，所以从A到E的距离是5+6=11。把这一信息刷新到表中。



重复步骤3和步骤4的操作。

7、遍历距离表找到从A出发距离最短的点，即顶点D（ACB已经遍历过了，不需要考虑）。

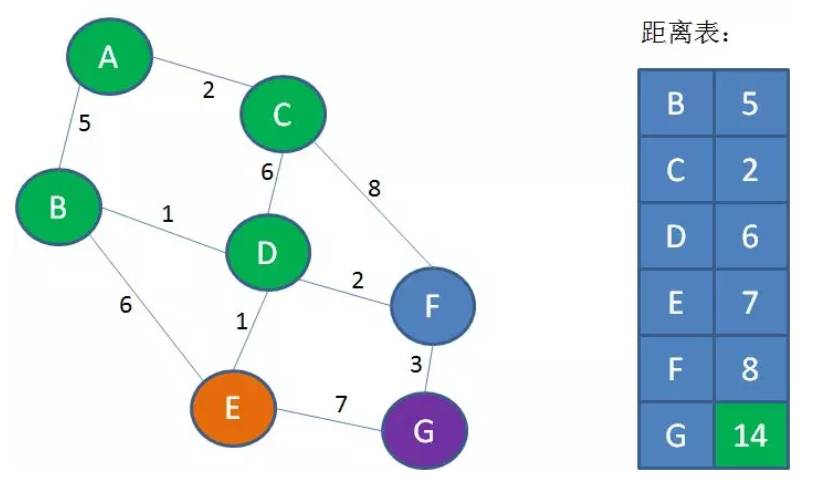
8、遍历顶点D，找到邻接点E和F。从D到E的距离是1，所以A到E的距离是6+1=7（小于距离表中的11）；从D到F的距离是2，所以从A到F的距离是6+2=8（小于距离表中的10）。把这一信息刷新到表中。



重复步骤3和步骤4的操作。

9、遍历距离表找到从A出发距离最短的点，即顶点E（ACBD已经遍历过了，不需要考虑）。

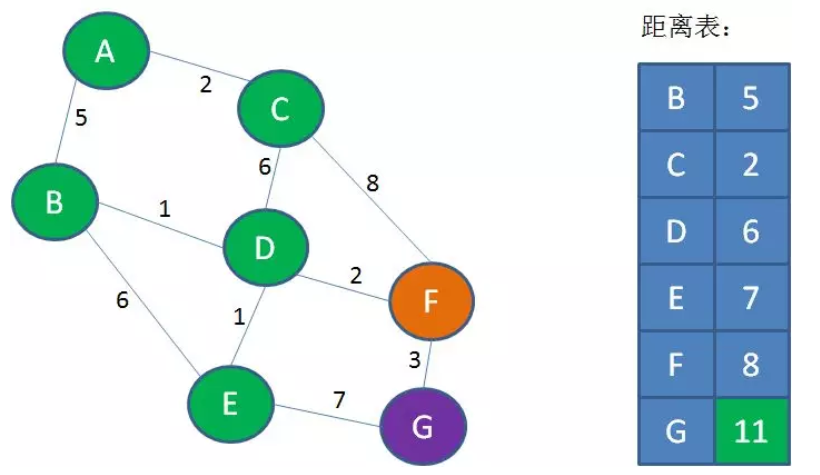
10、遍历顶点E，找到邻接点G。从E到G的距离是7，所以A到G的距离是7+7=14。把这一信息刷新到表中。



重复步骤3和步骤4的操作。

11、遍历距离表找到从A出发距离最短的点，即顶点F（ACBDE已经遍历过了，不需要考虑）。

12、遍历顶点F，找到邻接点G。从F到G的距离是3，所以从A到G的距离是8+3=11（小于距离表中的14）。把这一信息刷新到表中。



就这样除了顶点以外的全部顶点都已经遍历完毕，距离表存储的是从起点A到所有顶点的最短距离。显然从A到G的最短距离是11。（路径：A-B-D-F-G）