第二次作业

1. 设计一个高效算法，对实数线上给定的一个点集，求一个单位长度闭合区间的集合，包含所有给定的点，并要求此集合最小，证明你的算法的正确定。
2. 给定一个非负整数数组，你最初位于数组的第一个位置。数组中的每个元素代表你在该位置可以跳跃的最大长度。判断你是否能够到达最后一个位置。写出算法思路，并给出伪代码。

示例： 【2**, 3, 1, 1, 4】可达**

【3**, 2, 1, 0, 4**】不可达

答案：

1. 思路：先将点集以升序排列，从最小的点开始构建单位闭区间，其中包含所有与当前起点间距符合条件的点。下一个闭区间从没有被区间覆盖的所有点中的最小点开始，以此类推。

证明：？

1. 从第一个位置开始，用maxDistance保存当前可以到达的最远距离。每到达一个新的位置就判断其是否在maxDistance中，即是否可以作为起跳点。若可以起跳则进一步判断其落地位置是否在maxDistance之外，即maxDistance是否需要更新。

**maxDistance=num[0]**

//将maxDistance初始化为第0个位置的步长

**for i=1 to num.length**

//遍历各点

**if i<maxDistance**//如果从前面的点可以跳到当前点

**maxDistance=max(maxDistance,num[i]+i)**

//判断maxDistance是否需要更新

**if maxDistance>num.length**

//如果可以到达的最大距离大于最后一个点的位置则返回true

**return true**