

信封嵌套问题

分析：

1. 将数组按照宽度递增排序，当宽度相同时，按高度递减排序；
2. 将高度数组提取成一维数组，构成了一个一维数据求最长递增子序列长度的问题；

```
class Solution {
public:
    int maxEnvelopes(std::vector<std::vector<int>> &envelopes) {
        std::sort(envelopes.begin(),
                  envelopes.end(),
                  [](const std::vector<int> &a, const std::vector<int> &b) {
                      return a[0] != b[0] ? a[0] < b[0] : a[1] > b[1];
                  });

        std::vector<int> height;
        for (int i = 0; i < envelopes.size(); i++) {
            height.push_back(envelopes[i][1]);
        }

        return longofLIS(height);
    }

private:
    int longofLIS(std::vector<int> &height) {
        std::vector<int> dp(height.size() + 1, 1);

        for (int i = 1; i < height.size(); i++) {
            for (int j = 0; j < i; j++) {
                if (height[j] < height[i]) {
                    dp[i] = std::max(dp[i], dp[j] + 1);
                }
            }
        }

        int max = 0;
        for (auto item : dp) {
            max = (max > item ? max : item);
        }

        return max;
    }
};
```