

# 1578. Minimum Time to Make Rope Colorful

☰ Tags	medium
📄 Property	
🔗 URL	

## Question

原文：

Alice has `n` balloons arranged on a rope. You are given a **0-indexed** string `colors` where `colors[i]` is the color of the `i`th balloon.

Alice wants the rope to be **colorful**. She does not want **two consecutive balloons** to be of the same color, so she asks Bob for help. Bob can remove some balloons from the rope to make it **colorful**. You are given a **0-indexed** integer array `neededTime` where `neededTime[i]` is the time (in seconds) that Bob needs to remove the `i`th balloon from the rope.

Return the **minimum time** Bob needs to make the rope **colorful**.

我的理解：

給定一個字串 `colors` 一個陣列 `neededTime`，字串表示氣球顏色，陣列表示拿掉該氣球要花的時間，如果遇到相同顏色氣球連在一起的情況要拿掉同色氣球只拿一顆，並且要以最少的時間拿掉，也就是留下的那顆會是拿掉氣球要花的時間最長的那顆

return 拿掉氣球最短的總耗時

翻譯：

爱丽丝有 `n` 个气球排列在一条绳子上。给你一个**0索引**的字符串 `colors`，其中 `colors[i]` 是第 `i` 个气球的顏色。

爱丽丝希望绳子是色彩鮮艷的。她不希望**两个连续的气球**相同的顏色，所以她向鲍勃寻求帮助。鲍勃可以从绳子上取下一些气球，使其成为色彩鮮艷的。给你一个**0分的整数数组** `needTime`，其中 `needTime[i]` 是Bob需要从绳子上移走第 `i` 个气球的时间（秒）。

返回**鲍勃使绳子变色彩鮮艷所需的\*\*最小时间**。

自評翻譯正確性：

- Word Memory :

## Code

```
class Solution {
public:
    int minCost(string colors, vector<int>& neededTime) {
        //maxest mean the biggest neededTime value in that on loop
        //countTime mean total time for take out ballons
        int i=0,j=0,maxest=0,countTime=0;
        vector<int>temp;
        for(i=0;i<colors.size();i++){
            while(colors[i]==colors[i+1]){//if had two same color ballons
                if(neededTime[i]>maxest){//if neededTime of ballon bigger than maxest
                    maxest=neededTime[i];
                }
                if(neededTime[i+1]>maxest){
                    maxest=neededTime[i+1];
                }
                temp.push_back(neededTime[i]);//memory all of same color ballons
                if(colors[i+1]!=colors[i+2])
                    temp.push_back(neededTime[i+1]);
                i++;
            }
            for(j=0;j<temp.size();j++){//caculate time
                countTime=countTime+temp[j];
            }
            countTime=countTime-maxest;//take out the biggest one ballon neededtime
            temp.clear();
            maxest=0;
        }
        return countTime;
    }
};
```

思路：這題是要把如果鄰近的氣球是相同顏色，要把重複顏色的拿掉只留下一顆，但有一個限制條件是，拿掉氣球各自有不同的耗時，留下的一定要是耗時最長的那顆，我先把重複的氣球都先放進temp，同時間maxest紀錄裡面最大的一個neededtime是多少，之後先計算所有temp中的氣球的neededtime最後再減去我剛紀錄的最大的那一個needtime，然後繼續往後方的氣球找，如果又有找到相鄰且同色就再進入上面敘述的while，就達成最大的不

拿掉其餘拿掉的時間計算，有一些專門為剛好是相同氣球中的最後一個寫的例外程式碼，不然會少抓到最後一顆例如 紅色a的位置  
aaaabd

Success Details >

Runtime: 420 ms, faster than 15.95% of C++ online submissions for Minimum Time to Make Rope Colorful.

Memory Usage: 99.8 MB, less than 10.84% of C++ online submissions for Minimum Time to Make Rope Colorful.

Next challenges:

Group the People Given the Group Size They Belong To

Jump Game V

Average Height of Buildings in Each Segment

Show off your acceptance:



Time Submitted	Status	Runtime	Memory	Language
10/04/2022 21:07	Accepted	420 ms	99.8 MB	cpp

## 優良code參考

思路：