



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Experiment - 8

**Student Name:** Anshuman Singh

**UID:** 20BCS2665

**Branch:** CSE

**Section/Group:** 20BCS-DM-902/(A)

**Semester:** 6th

**Subject Code:** 20CSP-351

**Aim:** Can Place Flower Using

### Question :

You have a long flowerbed in which some of the plots are planted, and some are not. However, flowers cannot be planted in adjacent plots.

Given an integer array flowerbed containing 0's and 1's, where 0 means empty and 1 means not empty, and an integer n, return true if n new flowers can be planted in the flowerbed without violating the no-adjacent-flowers rule and false otherwise.

### CODE:

```
class Solution { public:  
  
    bool canPlaceFlowers(vector<int>& fd, int n) {  
        int pre = 1;  
        for(int i = 0; i < fd.size(); i++) {  
            if(pre == 1 && fd[i] == 0){  
                if(i+1 < fd.size() && fd[i+1] == 0){ pre = 0; n--;  
            }  
            else if(i+1 == fd.size()){ pre = 0; n--;  
            } } else if(fd[i] == 0) pre = 1; else pre = 0;  
        } return n <= 0;  
    }  
};
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## OUTPUT:

The screenshot shows a dark-themed interface of a coding platform. At the top, there's a green "Accepted" button with a circular arrow icon. Below it is a "Next question" link and a list of problems: "606. Construct String from Binary Tree" (selected), "495. Teemo Attacking", and "735. Asteroid Collision". Further down are buttons for "More challenges", "All statuses", and "All lang". At the bottom, a summary shows "Accepted" in green, "a few seconds ago", and "C++".

## Aim: Remove Duplicate Letters

Given a string  $s$ , remove duplicate letters so that every letter appears once and only once. You must make sure your result is the smallest in lexicographical order among all possible results.

## CODE:

```
class Solution { public:
    string removeDuplicateLetters(string s) {
        vector<int> v(26,0), vis(26,0);      for(const
        auto& it:s){  v[it-'a']++;
    }
    string res="";
    for(const auto& it:s){
        v[it-'a']--;
        if(!vis[it-'a']){
            while(res.size() > 0 && res.back() > it && v[res.back()-'a'] >0){
                vis[res.back()-'a']=0;
                res.pop_back();
            }
            res+=it;
            vis[it-'a']=1;
        }
    }
}
```

# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING



return res;

Discover. Learn. Empower.

## OUTPUT:

Description    🔒 Editorial    Solutions (1.2K)    Submissions

Accepted

Next question

- 317. Shortest Distance from All Buildings

More challenges

- 2030. Smallest K-Length Subsequence With Occurrences of a Letter

All statuses

Accepted    a few seconds ago

C++

