



# Homework3

23336003 陈政宇

## problem1

S = 'S1S2S3...Sn'，为一个长为n的字符串，存放在一个数组中，编写程序将S改造之后输出：

- 将S的所有第偶数个字符按照其原来的下标从大到小的次序放在S的后半部分。
- 将S的所有第奇数个字符按照其原来的下标从小到大的次序放在S的前半部分。

例如：S = 'ABCDEFGHijkl'，改造后的S为'ACEGIKLJHFDDB'

## solution1

```
void trans(string &str) {  
    int n = str.size();  
    string ptr;  
    for(int i = 1; i < n; i += 2) {  
        ptr += str[i];  
    }  
    for(int i = 2; i < n; i += 2) {  
        ptr += str[i];  
    }  
    str = ptr;  
}
```

## problem2

若s和t是用结点大小为1且不带头结点的单链表存储的两个字符串，设计算法，找出s中第一个不在t中出现的字符。

## solution2

```
struct node{
    char ch;
    node* next;
    node(char ch = 0, node* next = nullptr) : ch(ch), next(next) {}
};

char find(node *s, node *t) {
    auto it = t;
    bool lis[26] = {};
    while(it != nullptr) {
        lis[it -> ch - 'a'] = 1;
        it = it -> next;
    }
    it = s;
    while(it != nullptr) {
        if(!lis[it -> ch - 'a']) return it -> ch;
        it = it -> next;
    }
    return '\0';
}
```

## problem3

如果矩阵A中存在这样的—个元素A[i][j]满足条件：A[i][j]是第i行中值最小的元素，且又是第j列中值最大的元素，则可知该点为该矩阵的—个马鞍点。设计—个算法，计算出m x n的矩阵A的所有马鞍点。

## solution3

```
vector<pair<int, int>> calc(vector<vector<int>><int> a) {
    int n = a.size(), m = a[0].size();
    vector<int> mini(n), maxi(m);
    for(int i = 0; i < n; i++) mini[i] = INF;
    for(int i = 0; i < m; i++) maxi[i] = -INF;
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            mini[i] = min(mini[i], a[i][j]);
            maxi[j] = max(maxi[j], a[i][j]);
        }
    }
    vector<pair<int, int>> ans;
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            if(a[i][j] == mini[i] && a[i][j] == maxi[j])
                ans.push_back({i, j});
        }
    }
    return ans;
}
```