

# 奇偶划分

设计算法将整型数组A[n]中的元素调整为左右两部分

其中左边所有元素为奇数

右边所有元素为偶数

并要求算法的时间复杂度为 $O(n)$

```

#include <iostream>
#define MAXSIZE 10
using namespace std;
class SqList
{
public:
    //构造函数
    SqList(int m[])
    {
        for (int i = 0; i < MAXSIZE; i++)
        {
            data[i] = m[i];
        }
        length = MAXSIZE;
    }
    int length;
    int data[MAXSIZE];
};

int main(void)
{
    int num[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    SqList L1(num);
    int num2[MAXSIZE];
    SqList L2(num2);
    int m = 0, n = MAXSIZE;
    for (int i = 0; i < MAXSIZE; i++)
    {
        //从左赋值奇数
        if (L1.data[i] % 2 == 1)
        {
            L2.data[m] = L1.data[i];
            m++;
        }
        //从右赋值偶数
        else
        {
            L2.data[n - 1] = L1.data[i];
            n--;
        }
    }
    for (int i = 0; i < MAXSIZE; i++)
    {
        cout << L2.data[i] << ' ';
    }
    return 0;
}

```