

# C++方向编程题答案

## 第二周

### day8

题目ID: 45844 --两种排序方法

链接: <https://www.nowcoder.com/practice/839f681bf36c486fbcc5fcb977ffe432?tpId=85&&tqId=29844&rp=1&ru=/activity/oj&qu=/ta/2017test/question-ranking>

```
/*
    思路: 使用strcmp和strlen实现
*/

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int checkLexicograph(int n);
int checkLenth(int n);

char str[100][100];
int main()
{
    int n;
    int i;
    int lenSym, lexSym;

    scanf("%d", &n);
    for (i = 0; i < n; i++)
    {
        scanf("%s", str[i]);
    }
    lenSym = checkLenth(n);
    lexSym = checkLexicograph(n);
    if ((lenSym == 0) && (lexSym == 0))
    {
        printf("both\n");
    }
    if ((lenSym == -1) && (lexSym == 0))
    {
        printf("lexicographically\n");
    }
    if ((lenSym == 0) && (lexSym == -1))
    {
        printf("lengths\n");
    }
    if ((lenSym == -1) && (lexSym == -1))
    {
        printf("none\n");
    }
}
```

```

    return 0;
}
int checkLexicograph(int n)
{
    int i;
    for (i = 1; i < n; i++)
    {
        //strcmp按ASCII比较,如果前一个字符串小于当前字符串,则是字典序
        //如果大于,则不是
        if (strcmp(str[i-1],str[i]) > 0)
        {
            return -1;
        }
    }
    return 0;
}
int checkLenth(int n)
{
    int i;
    for (i = 0; i < n; i++)
    {
        if (strlen(str[i]) < strlen(str[i-1]))
        {
            return -1;
        }
    }
    return 0;
}

```

36932-求最小公倍数

<https://www.nowcoder.com/practice/22948c2cad484e0291350abad86136c3?tpId=37&ttId=21331&rp=1&ru=/activity/oj&ru=/ta/huawei/question-ranking>

```

/*
最小公倍数 = 两数之积除以最大公约数
*/
#include<iostream>
using namespace std;
/*
最大公约数使用辗转相除法: 即a与b的最大公约数可以转化为
a、b之间的余数为两者之间最小的数之间的公约数
*/
int gcd(int a, int b)
{
    int r;
    while(r = a%b){
        a = b;
        b = r;
    }
    return b;
}

```

```
int main()
{
    int a,b;
    while(cin >> a >> b){
        cout << a*b/gcd(a,b) <<endl;
    }
    return 0;
}
```

比特科技制作