

中国矿业大学计算机学院实验报告

课程名称	高级语言程序设计		实验名称	高级语言程序设计实践	
班级	信息安全 2019-1 班	姓名	李春阳	学号	10193657
仪器组号			实验日期	2020. 11. 24	

实验报告要求：1. 实验目的 2. 实验内容（题目描述，源代码，运行截图，调试情况） 3. 实验体会

一、实验目的

1. 认识了解 c++函数语法。
2. 掌握条件语句和循环语句。
3. 逐步学习算法内容，增加代码量。

二、实验内容

1、第一题

1.1 题目描述

A. 问题描述

编程序，实现如下功能：

- （1）定义两个一维数组 x, y, 不超过 50 个元素。
- （2）从键盘输入 k 个整数到数组 x 中。
- （3）计算 x 中数据的平均值 ave 及大于平均值的元素个数 n 并输出。
- （4）将数组 x 中数据复制到数组 y 中，重复的数据只存储一次，最后输出 y 中的数据。

B. 输入

输入两行，第一行为输入的数据个数，第二行若干整数，每个整数后面有一个空格用于分隔；

C. 输出

第一行若干整数，每个整数后面有一个空格，最后一个空格后换行；第二行若干整数，每个整数后面有一个空格，最后一个空格后不需要换行；

1.2 源代码

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    int x[50], y[50];
    int n = 0, k = 0;
    int i, j, l = 0;
```

```

double sum = 0, average = 0;
cin >> k;
for (i = 0; i < k; i++)
{
    cin >> x[i];
    sum += x[i];
}
average = sum / k;
for (i = 0; i < k; i++)
{
    if (x[i] > average)
    {
        n++;
    }
}
for (i = 0; i < k; i++)
{
    for (j = 0; j < i; j++)
    {
        if (x[i] == x[j])
        {
            break;
        }
    }
    if (i == j)
    {
        y[l] = x[i];
        l++;
    }
}

cout << average << " " << n << endl;
for (i = 0; i < l; i++)
{
    cout << y[i] << " ";
}
cout << endl;
return 0;
}

```

1.3 运行截图

```
Microsoft Visual Studio 调试控制台
6
6 3 4 3 2 9
4 5 2
6 3 4 2 9
E:\c++实践课作业\ConsoleApplication1\Debug\ConsoleApplication1.exe (进程 22920) 已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口。 . . .
```

1.4 调试情况

2、第二题

2.1 题目描述

A. 问题描述、

有 12 人围坐成一圈玩报数游戏，从 1 号人员开始顺时针报数，报到 k 的人员被淘汰出局；接着仍沿顺时针方向从被淘汰出局者的下一人员又重新从 1 开始报数，报到 k 的人被淘汰；如此继续，直到最后只剩下一个人时停止。请编写程序输出最后所剩那个人的编号。

注意：（1）假设参加游戏的人的编号沿顺时针方向依次为 1 到 12，可以使用数组来存放各数据；

（2） $k > 1$ ，由用户通过 `cin` 输入指定。

B. 输入

输入一个整数，代表报数值；

C. 输出

输出一个整数，即最后剩下的人的编号；

2.2 源代码

```
#include <iostream>
#include <cmath>

using namespace std;

int lastRemaining(int n, int m)
{
    int f = 0;
    for (int i = 2; i != n + 1; ++i)
    {
        f = (m + f) % i;
    }
}
```

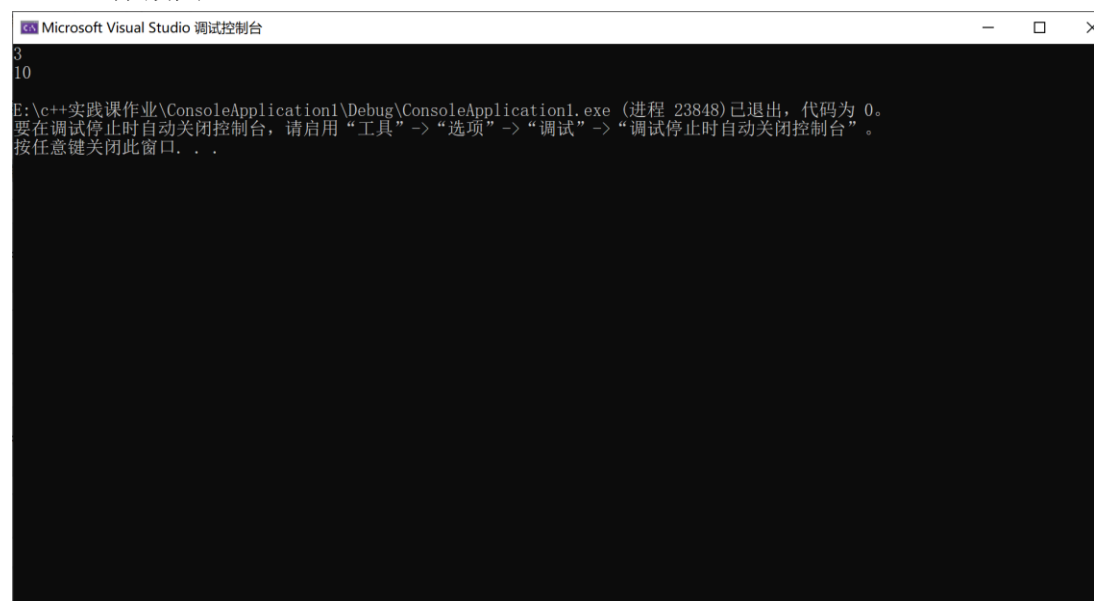
```

    }
    return f;
}

int main()
{
    int n, result;
    cin >> n;
    result = lastRemaining(12, n);
    cout << result + 1 << endl;
    return 0;
}

```

2.3 运行截图



2.4 调试情况

3、第三题

3.1 题目描述

A. 问题描述

小宗想知道两个日期之间所间隔的天数，他希望有一个日期计算器，输入两个日期后能够自动计算之间的天数。要求：设计相应的函数完成天数的计算，在主函数中验证正确性。

B. 输入

按照年月日的顺序输入两个日期，年月日之间用一个空格分隔；

C. 输出

输出两个日期之间的天数，即一个整数，整数后不需要换行；

3.2 源代码

```

#include <iostream>
#include <cmath>
using namespace std;
bool leap_year(int year)

```

```

{
    return ((year % 400 == 0) || (year % 100 != 0 && year % 4 == 0));
}

int date_to_int(int year, int month, int day)
{
    int month_length[] = { 0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    int ans = day - 1;
    while (month != 0) {
        --month;
        ans += month_length[month];
        if (month == 2 && leap_year(year))
            ans += 1;
    }
    ans += 365 * (year - 2000);
    ans += (year - 1) / 4 - 1971 / 4;
    ans -= (year - 1) / 100 - 1971 / 100;
    ans += (year - 1) / 400 - 1971 / 400;
    return ans;
}

struct Date
{
    int year;
    int month;
    int day;
};

int main()
{
    Date a, b;
    cin >> a.year >> a.month >> a.day;
    cin >> b.year >> b.month >> b.day;
    cout << abs(date_to_int(a.year, a.month, a.day) - date_to_int(b.year, b.month,
b.day));
    return 0;
};

```

3.3 运行截图

```
Microsoft Visual Studio 调试控制台
2016 3 6
2017 1 1
301
E:\c++实践课作业\ConsoleApplication1\Debug\ConsoleApplication1.exe (进程 24016) 已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口。 . . .
```

3.4 调试情况

4、第四题

4.1 题目描述

A. 问题描述

对于整型数组 **a**[10]和 **b**[10]，编制程序完成下列任务：

- (1) 由用户从键盘为两个数组输入值；
- (2) 求出两个数组的最大值和最小值；
- (3) 把数组 **a** 和 **b** 中的整数分别从小到大和从大到小排序；
- (4) 把两个有序的数组 **a** 和 **b** 组成一个长度为 20 的有序数组 **c**[20]，使数组 **c** 的顺序为从小到大。

B. 输入

输入两行整数，每行 10 个，第一行是数组 **a** 里的数组，第二行是数组 **b** 里的数值；

C. 输出

输出五行，第一行有两个整数，分别是数组 **a** 的最大值和最小值，两个整数之间用一个空格分隔；第二行有两个整数，分别是数组 **b** 的最大值和最小值，两个整数之间用一个空格分隔；第三行按照从小到大的顺序输出数组 **a** 里的数值，每个数字后面有一个空格，最后一个数字后面也有空格；第四行按照从大到小的顺序输出数组 **b** 里的数值，每个数字后面有一个空格，最后一个数字后面也有空格；第五行按照从小到大的顺序输出合并后数组 **c** 里的数值，每个数字后面有一个空格，最后一个数字后面也有空格。

4.2 源代码

```
#include <iostream>
#include <cmath>
#include <cstring>

using namespace std;
```

```
int max(int n[])
{
    int max = n[0];
    for (int i = 0; i < 10; i++)
    {
        if (max < n[i])
        {
            max = n[i];
        }
    }
    return max;
}

int min(int n[])
{
    int min = n[0];
    for (int i = 0; i < 10; i++)
    {
        if (min > n[i])
        {
            min = n[i];
        }
    }
    return min;
}

void maoPaoSort1(int array[], int len)
{
    for (int i = 0; i < len - 1; i++)
    {
        for (int j = 0; j < len - i - 1; j++)
        {
            if (array[j] > array[j + 1])
            {
                int temp;
                temp = array[j];
                array[j] = array[j + 1];
                array[j + 1] = temp;
            }
        }
    }
}
```

```

int main()
{
    int a[10], b[10], c[20];
    int i;
    for (i = 0; i < 10; i++)
    {
        cin >> a[i];
    }
    for (i = 0; i < 10; i++)
    {
        cin >> b[i];
    }
    for (i = 0; i < 10; i++)
    {
        c[i] = a[i];
        c[i + 10] = b[i];
    }
    maoPaoSort1(a, 10);
    maoPaoSort1(b, 10);
    maoPaoSort1(c, 20);
    cout << max(a) << " " << min(a) << endl;
    cout << max(b) << " " << min(b) << endl;
    for (i = 0; i < 10; i++)
    {
        cout << a[i] << " ";
    }
    cout << endl;
    for (i = 9; i > -1; i--)
    {
        cout << b[i] << " ";
    }
    cout << endl;
    for (i = 0; i < 20; i++)
    {
        cout << c[i] << " ";
    }
    cout << endl;
    return 0;
}

```

4.3 运行截图


```
Microsoft Visual Studio 调试控制台
2 5 9 1 3 4 0 6 7 8
10 5 25 9 6 3 7 1 2 13
9 0
25 1
0 1 2 3 4 5 6 7 8 9
25 13 10 9 7 6 5 3 2 1
0 1 1 2 2 3 3 4 5 5 6 6 7 7 8 9 9 10 13 25

E:\c++实践课作业\ConsoleApplication1\Debug\ConsoleApplication1.exe (进程 11004) 已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口。...
```

4.4 调试情况

5、第五题

5.1 题目描述

A. 问题描述

利用 `cin.getline()` 函数从键盘录入一句英文，其中每个单词之间用一个空格隔开，最后用 `!` 结束。统计该句话中单词的个数，并依次输出每个单词。输出个数后换行，输出每个单词后也换行。注意：在本平台下，`cin.getline()` 函数的使用方式如下：

```
char s[50];
```

```
cin.getline(s,50); //最多存储 49 个字符
```

B. 输入

输入一句英文，其中每个单词之间用一个空格隔开，最后一个单词后面用英文的 `!` 作为结束；

C. 输出

输出每个单词后换行，最后一行输出单词的数量。

5.2 源代码

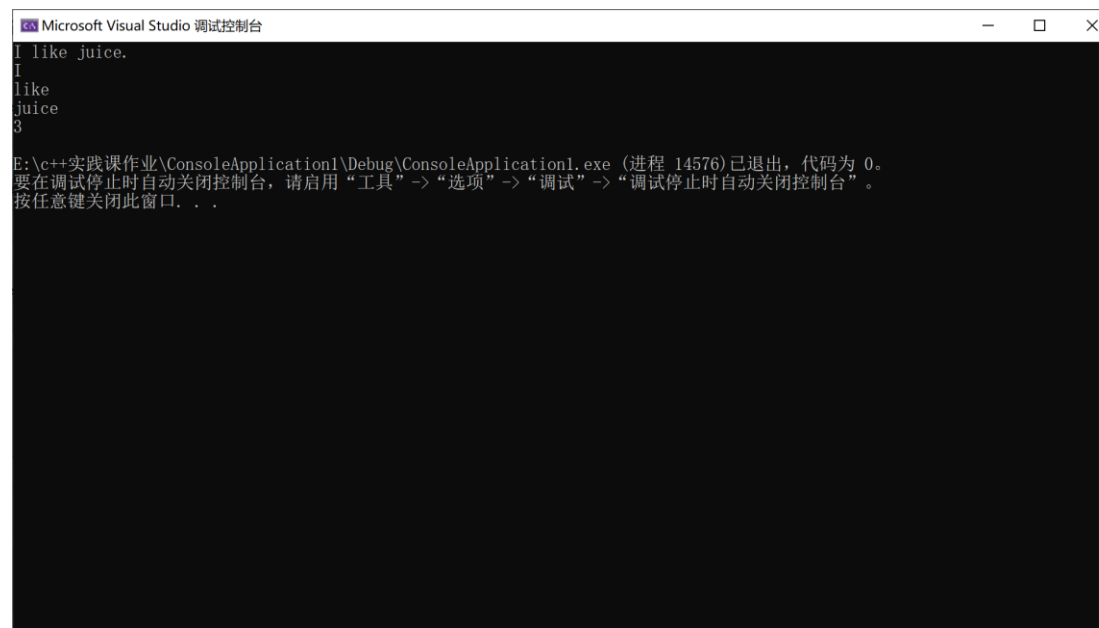
```
#include <iostream>
#include <cmath>
#include <cstring>
using namespace std;
int main()
{
    int num = 0, i;
    char str[50];
    cin.getline(str, 50, '.');
    for (i = 1; i < strlen(str); i++) {
        if (str[i] != ' ' && str[i - 1] == ' ')
        {
```

```

        num++;
    }
}
if (str[0] != ' ') {
    num++;
}
for (i = 0; i < strlen(str); i++)
{
    if (str[i] == ' ')
    {
        cout << endl;
    }
    else
    {
        cout << str[i];
    }
}
cout << endl;
cout << num << endl;
return 0;
}

```

5.3 运行截图



5.4 调试情况

三、实验体会

通过这次实验,我更加明白了 c++理论中的一些编程规范和 c++语言特性,掌握了基本编程知识,以后会更加认真的学习 c++理论知识,并不断实践和练习,在 debug 中不断学习。

