

《面向对象程序设计语言》作业（3.26）

地信班 109092023XXX 许愿

7. 找出一个二维数组中的鞍点，即该位置上的元素在该行上最大，在该列上最小（也可能没有鞍点）。

```
#include <iostream>
using namespace std;
int main(){
    const int n = 5, m = 5; // 行数和列数
    int a[n][m] =
    {{3,2,4,8,9},{1,2,3,4,5},{6,7,8,9,10},{11,12,13,14,15},{1,5,7,3,5}}; //
    定义数组

    int i, j, k, l; // 列号存储
    int max, min;
    for(i = 0; i < n; i++){ // 行内
        max = a[i][0]; // 初始化最大值为第一列的值
        k = 0;
        for(j = 1; j < m; j++){ // 列内
            if(a[i][j] > max){ // 若有比当前最大值大的值
                max = a[i][j]; // 更新最大值
                k = j;
            }
        }
        // 最终得到本列最大值为 max，列号为 k
        min = a[0][k];
        l = 0;
        for(j = 1; j < n; j++){
            if(a[j][k] < min){
                min = a[j][k];
                l = j;
            }
        }
        // 最终得到本行最小值为 min，行号为 l

        // 判断是否为鞍点
        if(l == i){ // 同时满足最大和最小
            cout << "鞍点为: " << a[i][k] << ", 位置为: " << i << ", " << k <<
endl;
            return 0;
        }
    }
```

```

    }
    cout << "没有鞍点" << endl; // 没有鞍点
    return 0;
}

PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work> & '
din=Microsoft-MIEngine-In-d5jehebn.ru4' '--stdout=Micr
zs.4mz' '--dbgExe=D:\msys64\ucrt64\bin\gdb.exe' '--int
● 鞍点为: 5, 位置为: 1,4
○ PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work>

```

8. 有 15 个数按从大到小顺序存放在一个数组中，输入一个数，要求用折半查找法找出该数是数组中第几个元素的值。如果该数不在数组中，则打印出“无此数”。

```

#include <iostream>
using namespace std;
int main(){
    int a[15] = {99, 88, 77, 66, 55, 44, 33, 22, 11, 10, 9, 8, 7, 6, 5}; //
    定义数组
    int num; // 输入的数
    cout << "请输入一个数: ";
    cin >> num;

    int left = 0;
    int right = 14; // 15-1

    while(left <= right){ // 左边界小于等于右边界
        int mid = (left + right) / 2; // 取出中间位置
        if(a[mid] == num){ // 若找到数
            cout << "该数在数组中的位置为: " << mid + 1 << endl; // 输出数在
            数组中的位置
            return 0;
        } else if(a[mid] < num){ // 中间值小于输入的数，左半部分查找
            right = mid - 1;
        } else { // 中间值大于输入的数，右半部分查找
            left = mid + 1;
        }
    }
    cout << "无此数" << endl; // 没有找到
    return 0;
}

```

```

PS C:\Users\Xuan\Desktop\24-
din=Microsoft-MIEngine-In-tv
w1.zgh' '--dbgExe=D:\msys64\
请输入一个数: 76
无此数
PS C:\Users\Xuan\Desktop\24-
din=Microsoft-MIEngine-In-3z
lg.iak' '--dbgExe=D:\msys64\
请输入一个数: 77
该数在数组中的位置为: 3
PS C:\Users\Xuan\Desktop\24-
din=Microsoft-MIEngine-In-vt
gz.0el' '--dbgExe=D:\msys64\
请输入一个数: 7
该数在数组中的位置为: 13

```

9. 给出年、月、日，计算该日是该年的第几天。

```

#include <iostream>
using namespace std;
int main(){
    int year, month, day; // 年、月、日
    cout << "请输入年、月、日: ";
    cin >> year >> month >> day;

    int days = 0; // 初始化天数

    for(int i = 1; i < month; i++){ // 遍历月份
        switch(i){ // 判断月份
            case 1: case 3: case 5: case 7: case 8: case 10: case 12: // 大
月
                days += 31;
                break;
            case 4: case 6: case 9: case 11: // 小月
                days += 30;
                break;
            case 2: // 2月特殊判断
                if((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)){ //
闰年的2月为29天
                    days += 29;
                } else { // 否则为2月28天
                    days += 28;
                }
                break;
        }
    }
    days += day; // 加上当前月已过的天数
}

```

```

    cout << "该日是该年的第" << days << "天" << endl; // 输出结果
    return 0;
}

```

```

PS C:\Users\Xuan\Desktop\24-25-2
din=Microsoft-MIEngine-In-cx00ie
li.0k3' '--dbgExe=D:\msys64\ucrt
请输入年、月、日: 2025 3 29
该日是该年的第88天
PS C:\Users\Xuan\Desktop\24-25-2
din=Microsoft-MIEngine-In-bfu5ck
ci.ynp' '--dbgExe=D:\msys64\ucrt
请输入年、月、日: 2024 3 29
该日是该年的第89天
PS C:\Users\Xuan\Desktop\24-25-2
din=Microsoft-MIEngine-In-tovo2f
sm.juq' '--dbgExe=D:\msys64\ucrt
请输入年、月、日: 2020 3 29
该日是该年的第89天
PS C:\Users\Xuan\Desktop\24-25-2

```

10. 有一篇文章，共有 3 行文字，每行有 80 个字符。要求分别统计出其中英文大写字母、小写字母、数字、空格及其他字符的个数。

```

#include <iostream>
#include <cstring>
using namespace std;
int main(){
    string data =
"BdOFkx*)KicbHD#XS$cCUMl$B~B+cXe BZ*NO}%FArptY/U/ujiWQdaK%t
&XwwHZ?w_Z=[=/%YFQ,V\n<RO)LnK/|fNb1SjyezxDN.cze{G+l<qET,G@X%?CwS&Eoh@N/
MbiurTiab: !+eNk-, {W[pibZ!Ldt+>E\n--FrZkHPrTJ^I~v?AdmFVg@B/Wls~<;yl@T/{W
/Ia(v#|UoVUnApXU:mhh)/&GkTj^VFYo|/)a?wKK^v";
    int cnt_dxzm, cnt_xxzm, cnt_sz, cnt_gk, cnt_qtzf = 0;
    for(int i = 0; i < data.length(); i++){
        if(data[i] >= 'A' && data[i] <= 'Z'){ // 大写字母
            cnt_dxzm++;
        } else if(data[i] >= 'a' && data[i] <= 'z'){ // 小写字母
            cnt_xxzm++;
        } else if(data[i] >= '0' && data[i] <= '9'){ // 数字
            cnt_sz++;
        } else if(data[i] == ' '){ // 空格
            cnt_gk++;
        } else { // 其他字符
            cnt_qtzf++;
        }
    }
}

```

```

}
cout << "大写字母个数: " << cnt_dxzm << endl;
cout << "小写字母个数: " << cnt_xxzm << endl;
cout << "数字个数: " << cnt_sz << endl;
cout << "空格个数: " << cnt_gk << endl;
cout << "其他字符个数: " << cnt_qtzf << endl;
return 0;
}

```



```

PS C:\Users\Xuan\Desktop\24
din=Microsoft-MIEngine-In-w
va.pjl' '--dbgExe=D:\msys64
大写字母个数: 85
小写字母个数: 131
数字个数: 0
空格个数: 3
其他字符个数: 76
PS C:\Users\Xuan\Desktop\24

```

11. 有一行电文，已按下面规律译成密码：A-Z, a-z, B-Y, b-y, C-X, c-x, ..
即第 1 个字母变成第 26 个字母，第 i 个字母变成第(26-i+1)个字母...
非字母字符不变。要求程序将密码译回原文，并打印出密码和原文。

```

#include <iostream>
using namespace std;
int main(){
    string dwmm =
"wfdaujfvgaYWGDAydgawesuydhQQSDYAwyuixdahuicha283164r2fcgvy8xd8swacy7
";
    string dw = "";

    for(int i = 0; i < dwmm.length(); i++){
        if(dwmm[i] >= 'A' && dwmm[i] <= 'Z'){ // 大写字母
            dw += 'A' + ('Z' - dwmm[i]);
        } else if(dwmm[i] >= 'a' && dwmm[i] <= 'z'){ // 小写字母
            dw += 'a' + ('z' - dwmm[i]);
        } else { // 非字母字符
            dw += dwmm[i];
        }
    }
    cout << "密码是: " << dwmm << endl;
    cout << "原文是: " << dw << endl;
    return 0;
}

```

```
PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work> & 'c:\Users\Xuan\.vscode\extensions\Microsoft-MIEngine-In-j4ylggxo.mbj' '--stdout=Microsoft-MIEngine-Out-xmguuyo0.y
ni.ukn' '--dbgExe=D:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
● 密码是: wfdaujfvgaYWGADYdgawesuydhQQSDYAwyhuixdahuicha283164r2fcgvy8xd8swacy7
原文是: duwzfquetzBDTWZwbwtzdvhfbwsJJHwBZdbsfrcwzsfrcsz283164i2uxteb8cw8hdzxb7
```

14. 输入 n 个字符串，将它们按字母由小到大的顺序排列并输出。

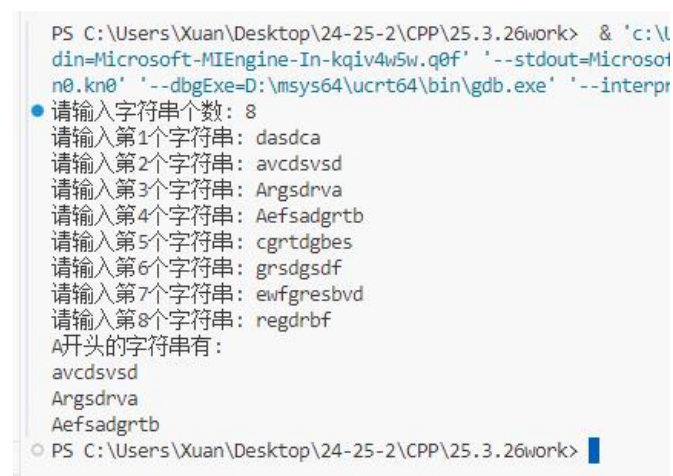
```
#include <iostream>
using namespace std;
int main(){
    int n;
    cout << "请输入要输入的字符串数量: ";
    cin >> n;
    string* str = new string[n];
    for(int i = 0; i < n; i++){
        cout << "请输入第" << i + 1 << "个字符串: ";
        cin >> str[i];
    }
    // 排序
    for(int i = 0; i < n - 1; i++){
        for(int j = 0; j < n - i - 1; j++){
            if(str[j] > str[j + 1]){
                string temp = str[j];
                str[j] = str[j + 1];
                str[j + 1] = temp;
            }
        }
    }
    // 输出
    cout << "排序后的字符串为: " << endl;
    for(int i = 0; i < n; i++){
        cout << str[i] << endl;
    }
    return 0;
}
```

```
PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work> & 'c:\Users\Xuan\.vscode\extensions\Microsoft-MIEngine-In-xezvanrm.ikf'
ex.asi' '--dbgExe=D:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
● 请输入要输入的字符串数量: 3
  请输入第1个字符串: awsdascascv
  请输入第2个字符串: fegtdfbvsda
  请输入第3个字符串: cefgrdsbhgd
  排序后的字符串为:
  awsdascascv
  cefgrdsbhgd
  fegtdfbvsda
```

15. 输入 n 个字符串，把其中以字母 A 打头的字符串输出。

```
#include <iostream>
using namespace std;
int main(){
    int n = 0;
    cout << "请输入字符串个数: ";
    cin >> n;
    string* str = new string[n]; // 生成字符串数组
    for(int i = 0; i < n; i++){
        cout << "请输入第" << i + 1 << "个字符串: ";
        cin >> str[i]; // 输入字符串
    }

    cout << "A 开头的字符串有: " << endl;
    for(int i = 0; i < n; i++){
        if(str[i][0] == 'A' || str[i][0] == 'a'){ // 首字母开头为 A 或 a
            cout << str[i] << endl;
        }
    }
    return 0;
}
```



```
PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work> & 'c:\U
din=Microsoft-MIEngine-In-kqiv4w5w.q0f' '--stdout=Microsof
n0.kn0' '--dbgExe=D:\msys64\ucrt64\bin\gdb.exe' '--interpr
● 请输入字符串个数: 8
  请输入第1个字符串: dasdca
  请输入第2个字符串: avcdsvsd
  请输入第3个字符串: Argsdrva
  请输入第4个字符串: Aefsadgrtb
  请输入第5个字符串: cgrtdgbes
  请输入第6个字符串: grsdgsdf
  请输入第7个字符串: ewfgresbvd
  请输入第8个字符串: regdrbf
  A开头的字符串有:
  avcdsvsd
  Argsdrva
  Aefsadgrtb
○ PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work>
```

17. 输入 10 个学生的姓名、学号和成绩，将其中不及格者的姓名、学号和成绩输出。

```
#include <iostream>
using namespace std;
int main(){
    const int n = 10; // 总共 10 名学生
    string* name = new string[n]; // 姓名
    string* id = new string[n]; // 学号
```



```

int* score = new int[n]; // 成绩

for(int i = 0; i < n; i++){
    cout << "请输入第" << i + 1 << "个学生的姓名: ";
    cin >> name[i]; // 输入姓名
    cout << "请输入第" << i + 1 << "个学生的学号: ";
    cin >> id[i]; // 输入学号
    cout << "请输入第" << i + 1 << "个学生的成绩: ";
    cin >> score[i]; // 输入成绩
}

cout << "不及格的学生有: " << endl;
for(int i = 0; i < n; i++){
    if(score[i] < 60){ // 成绩不及格
        cout << "姓名: " << name[i] << ", 学号: " << id[i] << ", 成绩: "
        << score[i] << endl;
    }
}
return 0;
}

```

```

PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work> & 'c:\Us
din=Microsoft-MIEngine-In-apu3x1q5.iga' '--stdout=Microsoft
bz.oao' '--dbgExe=D:\msys64\ucrt64\bin\gdb.exe' '--interpre

```

```

● 请输入第1个学生的姓名: 小兰
  请输入第1个学生的学号: 1
  请输入第1个学生的成绩: 99
  请输入第2个学生的姓名: 小红
  请输入第2个学生的学号: 2
  请输入第2个学生的成绩: 99
  请输入第3个学生的姓名: 小绿
  请输入第3个学生的学号: 3
  请输入第3个学生的成绩: 33
  请输入第4个学生的姓名: 小郑
  请输入第4个学生的学号: 5
  请输入第4个学生的成绩: 55
  请输入第5个学生的姓名: 小刘
  请输入第5个学生的学号: 7
  请输入第5个学生的成绩: 77
  请输入第6个学生的姓名: 小黄
  请输入第6个学生的学号: 4
  请输入第6个学生的成绩: 33
  请输入第7个学生的姓名: 小李
  请输入第7个学生的学号: 6
  请输入第7个学生的成绩: 88
  请输入第8个学生的姓名: 小钱
  请输入第8个学生的学号: 10
  请输入第8个学生的成绩: 29
  请输入第9个学生的姓名: 小芳
  请输入第9个学生的学号: 33
  请输入第9个学生的成绩: 100
  请输入第10个学生的姓名: 小邓
  请输入第10个学生的学号: 60
  请输入第10个学生的成绩: 21433542
  不及格的学生有:
  姓名: 小绿, 学号: 3, 成绩: 33
  姓名: 小郑, 学号: 5, 成绩: 55
  姓名: 小黄, 学号: 4, 成绩: 33
  姓名: 小钱, 学号: 10, 成绩: 29

```

```

○ PS C:\Users\Xuan\Desktop\24-25-2\CPP\25.3.26work>

```