

字符串

人员

韩承睿、牟茗、辛帅辰、高健桓、方俊喆、崔吉诺、刘祺、齐振玮、谢亚锴、王静嘉、牛同泽、徐浩然 到课；李翰如、秦显森 线上

刘嘉航、刘峰铄 未到

作业

作业：noi 1.7 03

要求下节课默写 noi 1.7 02

课堂表现

有些同学上课第一遍没有听明白，一定要及时给老师反馈。

要不然老师以为同学们都听懂了，就会继续往下讲，同学们就更容易落下了

课堂内容

东方博宜 1327. 鲜花方阵

```

#include <iostream>

using namespace std;

int a[15][15];

int main() {
    int n; cin >> n;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            a[i][j] = 1;
        }
    }

    int x = (n+1) / 2;

    a[x][x] = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (i+j==x+1 || i+j==x+n || i-j==x-1 || i-j==x-n) {
                a[i][j] = 0;
            }
        }
    }

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            printf("%3d", a[i][j]);
        }
        cout << endl;
    }
    return 0;
}

```

字符串(字符数组)

```

#include <iostream>
#include <cstring> // strlen 的头文件

using namespace std;

char a[100];

int main()
{
    cin >> (a+1); // 从 a[1] 位置往后输入字符串，直接写 cin>> a 的话代表从 a[0] 往后输入
    int len = strlen(a+1); // 从 a[1] 位置往后求字符串的长度，遇到 '\0' 截止
    for (int i = 1; i <= len; i++) { // 遍历字符串的每一位
        cout << a[i] << endl;
    }
    cout << (a+1); // 从 a[1] 位置往后输出字符串
    return 0;
}

```

ASCII码的意思：

一个字符会对应着一个整数

例如：

```

'0' - 48
'A' - 65
'a' - 97

```

两个字符比较大下：比较的实际是两个字符对应的ascii码的大小

字符串比较大小：按照字典序的比较方式进行比较

若有两个字符串 s 和 t，从两个字符串第一个位置的字符开始比较

当两个字符串对应位置的字符不同时，哪个位置的字符小，则其对应的字符串小

若相同，继续往后比

例如：

```

"abc" < "bca"
"abc" < "ad"
"abc" < "abcd"

```

字符串的结尾：'\0'

noi 1.7 02:找第一个只出现一次的字符

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

using namespace std;

char s[100005];
int sum[150];

int main()
{
    cin >> (s+1);
    int len = strlen(s+1);
    for (int i = 1; i <= len; i++) {
        char c = s[i];
        sum[c]++;
    }
    bool flag = false;
    for (int i = 1; i <= len; i++) {
        char c = s[i];
        if (sum[c] == 1) {
            cout << c;
            flag = true;
            break;
        }
    }
    if (flag == false) {
        cout << "no";
    }
    return 0;
}
```