

选择排序

人员

韩承睿、牟茗、刘嘉航、辛帅辰、高健桓、李翰如、方俊喆、崔吉诺、刘祺、夏硕承、秦显森、齐振玮、谢亚锴、王静嘉、牛同泽、徐浩然、刘轩铜 到课

作业检查

韩承睿 已完成

牟茗 已完成

刘嘉航 未打卡

辛帅辰 已完成

高健桓 已完成

李翰如 已完成

方俊喆 已完成

崔吉诺 已完成

刘祺 已完成

夏硕承 已完成

秦显森 未打卡

齐振玮 未打卡

谢亚锴 已完成

王静嘉 未打卡

牛同泽 已完成

徐浩然 已完成

刘轩铜 未打卡

作业

必做：1453 - 橘子排队 和 1166 - 数的排序

选做：1326 - 需要安排几位师傅加工零件

下周默写：noi 1.7 31:字符串p型编码 和 1010 - 数组元素的排序

课堂表现

许多同学在课堂出怪声，听讲不认真，课堂纪律比较差。

同时有个别同学课上看视频被老师批评，希望以后改正。

课堂内容

noi 1.7 31:字符串p型编码

跟之前做过的一道题目类似，只需考虑跟前一个数是否一致，来记录目前连续的数的数量即可

```
// 方法一
#include <iostream>
#include <cstring>

using namespace std;

char s[1005];
int f[1005];

int main()
{
    cin >> (s+1);
    int n = strlen(s+1);

    for (int i=1; i<=n; i++) {
        if (s[i] == s[i-1]) {
            f[i] = f[i-1] + 1;
        } else {
            f[i] = 1;
        }
    }

    for (int i=2; i <= n; i++) {
        if (s[i] != s[i-1]) {
            cout << f[i-1] << s[i-1];
        }
    }
    cout << f[n] << s[n] << endl;
    return 0;
}
```

```
// 方法二
#include <iostream>
#include <cstring>

using namespace std;

char s[1005];

int main()
```

```

{
    cin >> (s+1);
    int n = strlen(s+1);

    int cnt = 0;
    for (int i=1; i<=n; i++) {
        if (s[i] == s[i-1]) {
            cnt++;
        } else {
            if (i-1 != 0) {
                cout << cnt << s[i-1];
            }
            cnt = 1;
        }
    }

    cout << cnt << s[n];
    return 0;
}

```

noi 1.7 10:简单密码

```

#include <iostream>
#include <cstring>
#include <string>

using namespace std;

int main()
{
    string s;
    getline(cin,s);
    int n = s.size();
    for (int i = 0; i < n; i++) {
        if (s[i]>='A'&& s[i]<='Z') {
            if (s[i]>='F') {
                s[i] -= 5;
            }
            else {
                s[i] += 21;
            }
        }
    }
    cout << s << endl;
    return 0;
}

```

```

// 方法二，比较不好理解的一种方法
#include <iostream>

```

```
#include <cstring>
#include <string>

using namespace std;

int main()
{
    string s;
    getline(cin,s);
    int n = s.size();
    for (int i = 0; i < n; i++) {
        if (s[i]>='A'&& s[i]<='Z') {
            s[i] = ((s[i]-'A'-5+26)%26) + 'A';
        }
    }
    cout << s << endl;
    return 0;
}
```

noi 1.7 21:单词替换

可以采用 string 数组的方式过

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

using namespace std;

string s[105];

int main()
{
    int n = 1;
    while (cin >> s[n]) {
        n++;
    }

    string a = s[n-2], b = s[n-1];
    for (int i = 1; i <= n-3; i++) {
        if (s[i] == a) {
            s[i] = b;
        }
    }

    for (int i = 1; i <= n-3; i++) {
        cout << s[i] << " ";
    }
    cout << endl;
}
```

```
    return 0;
}
```

东方博宜 1010 - 数组元素的排序 (选择排序)

先把最小的数放在第一个位置，然后把第二小的数放在第二个位置，以此类推.....

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

using namespace std;

int a[15];

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

    for (int i = 1; i <= n; i++) {
        for (int j = i+1; j <= n; j++) {
            if (a[j] < a[i]) {
                int t = a[i];
                a[i] = a[j];
                a[j] = t;
            }
        }
    }

    for (int i = 1; i <= n; i++) {
        cout << a[i] << " ";
    }
    cout << endl;
    return 0;
}
```