# 计算机和编程语言

翁恺

#### 计算机是生活必需品

- 如果想要自己设计一些功能和行为,让计算机按照你的意图做事情,就需要写程序了
- 本课程的主要的目的是通过学习编程,来理解计算机 是如何解决问题的,理解计算机的能与不能,擅长与 非擅长

• "请给我一杯水"

- "请给我一杯水"
  - 1. 转身走到厨房;

- "请给我一杯水"
  - 1. 转身走到厨房;
  - 2. 找到一个杯子;

- "请给我一杯水"
  - 1. 转身走到厨房;
  - 2. 找到一个杯子;
  - 3. 找到一个水壶;

- "请给我一杯水"
  - 1. 转身走到厨房;
  - 2. 找到一个杯子;
  - 3. 找到一个水壶;
  - 4. 在杯子中倒入一些水;

- "请给我一杯水"
  - 1. 转身走到厨房;
  - 2. 找到一个杯子;
  - 3. 找到一个水壶;
  - 4. 在杯子中倒入一些水;
  - 5. 拿着杯子走回桌子。

- "请给我一杯水"
  - 1. 转身走到厨房;

2. 找到一个杯子; 人: What to do 计算机: How to do

- 3. 找到一个水壶;
- 4. 在杯子中倒入一些水;
- 5. 拿着杯子走回桌子。

### 计算机语言

- 程序是用特殊的编程语言写出来表达如何解决问题的
- 不是用编程语言来和计算机交谈,而是描述要求它如何做事情的过程或方法

#### 计算机的语言

```
05 00 00 00
                                                     00 00 02 00
01 00 04 28
             49 29 56 00
                          21 00 04 00
                          08 00 00 00
01 00 06 00
             07 00 01 00
                                       1D 00 01 00
                                                    01 00 00 00
05 2A B7 00
             01 B1 00 00
                          00 01 00 09
                                       00 00 00 06
                                                     00 01 00 00
             00 0A 00 0B
                                       00 00 00 63
00 01 00 09
                          00 01 00 08
                                                     00 02 00 04
             10 20 3C 10
                          1A 3D 1C 99
                                                    70 3E 1C 3C
                                       00 0E 1B 1C
00 00 00 1D
1D 3D A7 FF
                          1B B6 00 03
                                       B1 00 00 00
             F4 BZ 00 0Z
                                                     02 00 09 00
00 00 26 00
             09 00 00 00
                          03 00 03 00
                                       04 00 06 00
                                                     05 00 0A 00
                          08 00 12 00
06 00 0E 00
             07 00 10 00
                                       09 00 15 00
                                                     0A 00 1C 00
             00 00 08 00
                          02 FD 00 06
                                       01 01 0E 00
0B 00 0C 00
                                                     01 00 0D 00
00 00 02 00
            0E
```

#### 计算机的语言

```
00 00 02 00
           49 29 56 00
01 00 04 28
                       21 00 04 00
                                   05 00 00 00
                       08 00 00 00
                                   1D 00 01 00
01 00 06 00
           07 00 01 00
                                               01 00 00 00
           01 B1 00 00
05 2A B7 00
                       00 01 00 09
                                    00 00 00 06
                                               00 01 00 00
           00 0A 00 0B
                       00 01 00 08
00 01 00 09
                                   oint u = 32;
           10 20 3C 10
                       1A 3D 1C 99
00 00 00 1D
           F4 B2 00 02
                                    Pint v = 26;
1D 3D A7 FF
                       1B B6 00 03
00 00 26 00
                       03 00 03 00
           09 00 00 00
                                    while (v!=0) {
           07 00 10 00
                       08 00 12 00
06 00 0E 00
                                          int temp = u\%v;
                       02 FD 00 06
           00 00 08 00
0B 00 0C 00
00 00 02 00
           0E
                                          u = V;
                                          v = temp;
                                    System.out.println(u);
```

# 辗转相除法

## 辗转相除法

```
int u = 32;
int v = 26;
while ( v!= 0 ) {
    int temp = u%v;
    u = v;
    v = temp;
}
System.out.println(u);
```

## 辗转相除法

```
int u = 32;
int v = 26;
while (v!=0)
   int temp = u\%v;
   u = V;
   v = temp;
System.out.println(u); 3. 回到第一步。
```

- 1. 如果v等于0, 计算结束, u就是最大公约数;
- 2. 如果v不等于0,那么计算u除以v的余数,让u 等于v,而v等于那个余数;

# 计算机-程序-算法

```
05 00 00 00
                                                  00 00 02 00
01 00 04 28
            49 29 56 00
                        21 00 04 00
                        08 00 00 00
                                     1D 00 01 00
01 00 06 00
            07 00 01 00
                                                  01 00 00 00
                        00 01 00 09
                                    00 00 00 06 00 01 00 00
05 2A B7 00
            01 B1 00 00
00 01 00 09
            00 0A 00 0B
                        00 01 00 08 00 00 00 63
                                                  00 02 00 04
                        1A 3D 1C 99 00 0E 1B 1C
                                                 70 3E 1C 3C
            10 20 3C 10
00 00 00 1D
                        1B B6 00 03 B1 00 00 00
1D 3D A7 FF
            F4 BZ 00 0Z
                                                  02 00 09 00
                                    04 00 06 00
00 00 26 00
            09 00 00 00
                        03 00 03 00
                                                  05 00 0A 00
            07 00 10 00 08 00 12 00 09 00 15 00
                                                 0A 00 1C 00
06 00 0E 00
           00 00 08 00 02 FD 00 06 01 01 0E 00
0B 00 0C 00
                                                 01 00 0D 00
00 00 02 00 0E
```

# 计算机-程序-算法

```
01 00 04 28
                                            00 00 02 00
           dint u = 32;
01 00 06 00
                                            01 00 00 00
05 2A B7 00 0 int V = 26;
                                            00 01 00 00
00 01 00 09 0while (v!= 0) {
                                            00 02 00 04
00 00 00 1D
                                           70 3E 1C 3C
                 int temp = u\%v;
1D 3D A7 FF F
                                            02 00 09 00
00 00 26 00 0 U = V;
                                            05 00 0A 00
               v = temp;
06 00 0E 00
                                            0A 00 1C 00
0B 00 0C 00
                                            01 00 0D 00
00 00 02 00
            System.out.println(u);
```

# 计算机-程序-算法

```
00 00 02 00
01 00 04 28
          dint u = 32;
01 00 06 00
                                          01 00 00 00
05 2A B7 00 0 int V = 26;
                                          00 01 00 00
          00 02 00 04
00 01 00 09
                                         70 3E 1C 3C
00 00 00 1D
                int temp = u\%v;
1D 3D A7 FF
                                          02 00 09 00
          u = V;
00 00 26 00
                                          05 00 0A 00
                                          0A 00 1C 00
06 00 0E 00
               v = temp;
0B 00 0C 00
00 00 02 00
           System.out.println(u); <sub>10,</sub> 那么计算u除以v的余数,让u等于v,而
```

计算结束, u就是最大公约数;

v等干那个余数;

3. 回到第一步。

### 算法

- 我们要让计算机做计算,就需要像这样找出计算的步骤,然后用编程语言写出来
- 计算机做的所有的事情都叫做计算
- 计算的步骤就是算法

• 2x+6=20

- 2x+6=20
  - 解方程

- 2x+6=20
  - 解方程
  - 枚举

- 2x+6=20
  - 解方程
  - 枚举
  - 二分搜索

### 枚举求最大公约数

- 1. 设t为2;
- 2. 如果u和v都能被t整除,则记下这个t
- 3. t加1后重复第2步,直到t等于u或v;
- 4. 那么,曾经记下的最大的可以同时整除u和v的t就是gcd

### 程序的执行

- 解释:借助一个程序,那个程序能试图理解你的程序,然后按照你的要求执行
- 编译:借助一个程序,就像一个翻译,把你的程序翻译成计算机真正能懂的语言——机器语言——写的程序,然后,这个机器语言写的程序就能直接执行了

### 解释语言vs编译语言

- 语言本无编译/解释之分
- 常用的执行方式而已
- 解释型语言有特殊的计算能力
- 编译型语言有确定的运算性能