

# Java互助课堂（周一

## 1.Introduction

徐延楷 a.k.a. Froster

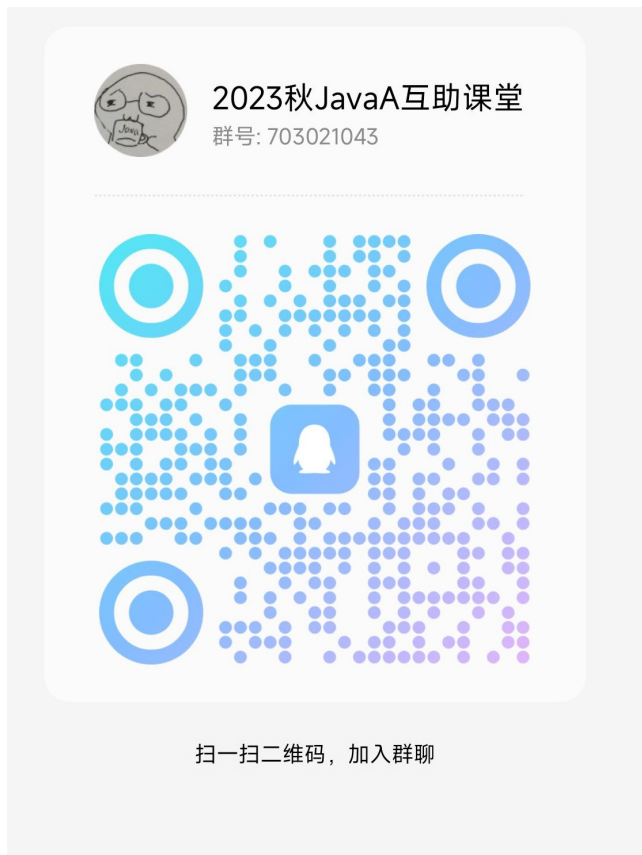
20级的老东西

本来想叫它Programming 101的，还是跟互助课堂吧

周一和周五的风格大概也许差别挺大

# 扫左边的签到，右边的加互助课堂群

(来这的不会有人没加互助课堂群吧orz)



# 课前声明

先水一下时间

我个人是第一次讲课，肯定不如讲了好多年的教授，所以大课还是要去的

课件是基于自己的理解做的，很可能充满知识的诅咒。如果没听懂或者觉得我讲错了，直接提问。

我就是个大四的老东西，水平很可能不如你们，和教授说话还犯社恐，所以放心讨论问题

前几届的互助课堂群的讨论和交流氛围挺融洽的，课下（或者课上）有问题欢迎往那里面发

- 但是别在里面发癫
- **更别在里面发大段代码，以及搞什么抄袭代写的，被抓到非常可能直接挂科**

# 为什么要写代码？

- 计算
  - 求解  $f(x) = ax + b$
  - 求解  $f(x) = 1145f(x^{14}) + 1919x - \frac{810}{x}$
  - 求解一张图像上有没有人脸
  - 求解从宝安机场到南科大怎么走最近
  - 现有一张excel表，过滤掉里面所有姓名不为张三的人
- 控制
  - 让一盏led灯亮起来
  - 让一盏led灯每秒闪烁一次
  - 让一盏led灯以1, 1, 4, 5, 1, 4秒的间隔循环闪烁
  - 现有一辆往前跑的小车，如果车前面有障碍物则右转继续跑
  - 让一个游戏人物的模型在按下鼠标左键时做铁山靠

# 功能的组合

如何实现上面的需求？

- 求解  $f(x) = 114x + 514$ 
  - 输入  $x$  （从设备，比如键盘，输入）
  - 计算乘法： $114 \times x$ ，**保存结果**，记作  $a$
  - 计算加法： $514 + a$ ，保存结果，记作  $b$
  - 输出  $b$  （显示到某个地方）

```
void f() {  
    // print(114 * readNumber() + 514);  
    int x = readNumber();  
    int a = 114 * x;  
    int b = a + 514;  
    print(b);  
}
```

做数学题的时候在草稿纸上会这么写：

$$\begin{aligned} 114 * 10 &= 1140 \\ 1140 + 514 &= 1654 \end{aligned}$$

算数的时候有点像工厂里的流水线——一个东西进来，先在前面的机械被加工。然后在后面被加工，以此类推...

代码的每一行就像是加工机械，“东西”是数据，你的一堆机械把数据加工成你想要的样子。

# 功能的组合

- 有一辆小车，让它持续前进，如果车前面有障碍物则右转再继续前进
  - 控制小车的轮子往前滚
  - 读取传感器数据
  - 如果：
    - 传感器数据显示有障碍：
      - 控制轮子停下
      - 右转
    - 传感器数据显示没有障碍：
      - 什么都不做
  - 回到第一步

```
void control() {  
    while (active) {  
        moveForward();  
        SensorData data = readSensorData();  
        if (hasObstacle(data)) {  
            stopMoving();  
            turnRight();  
        } else {  
            // do nothing  
        }  
    }  
}
```

# 功能的组合

编程的本质其一

rt, 组合已有的功能去完成你的需求。

最类似的可能是化学大题里写有机合成路线。

对已有功能的记忆+熟悉常见的组合模式。

# 功能从哪来？

layers of abstraction

计算机（在这门课里是jvm）提供：加减乘除，判断，跳转，存储...

编程语言（在这门课里是jre）和第三方库提供：乘方，对数，求导，读写文件，访问网络，控制外设...

编程的绝大部分工作是在发明自己的功能（我们把这个叫函数/方法）。

鲁迅没有说过：编程就是把问题不断分解为小问题，小问题分解为更小的问题，最终分解到已有的功能上去。

```
void anotherControl() {  
    boolean flag = isButtonPressed();  
    if (flag) {  
        control();  
    }  
}  
  
boolean isButtonPressed() {  
    ...  
}
```



# 来点练习?

regex!

## 正则表达式

一个匹配文本的工具。虽然不能像编程语言一样做很多事，但在查找和替换文本上极其有用。

### 1. 准备工作

- 打开你的idea/vscode...，如果你没有ide，打开<https://regex101.com/>
- 对于使用ide的同学，打开查找和替换功能（一般是ctrl+r）
- 在上方的查找框最右侧有一个 `.*` 图标。点击进入正则表达式搜索模式。
- 准备一段文本（这个之后我准备）

# 基础语法

省略版

普通字符：

- 普通字母数字和符号：匹配普通字母数字和符号。和平时的搜索功能一样，输入aa1可搜索出所有"aa1"。

特殊字符不能直接匹配，需要加\转义。如要匹配".", 搜索栏里对应\.

- .: 匹配任意字符。
- \w: 匹配单个字母    \w
- \d: 匹配单个数字
- \n: 匹配换行符

# 基础语法

省略版

特殊字符（对普通字符的修饰）：

- `*`：尽可能多次（0次也行）的匹配\*前面的单个字符。比如，`ca*bx`可以匹配"cbx", "cabx", "caaaaaaaabx"。  
`c.*b`匹配"cb", "caaaaabaaab", "c114514e5b"。
- `?`：
  - 如果前面不为`*`，匹配前面的字符0或1次。比如，`ca?b`匹配"cab"和"cb", `c.?b`匹配"cb", "c1b"...
  - 如果前面为`*`，则`*?`一起尽可能少的匹配\*前面的单个字符。对于"caaaaabaaab", `c.*b`匹配整个字符串，而 `c.*?b`匹配前半段"caaaaab"（遇到第一个b就停了）。`.*?`是一个很实用的组合。
- `[]`：匹配任意中括号内的字符。如`[ab]`匹配a或b。`[\w\n]`匹配单个字母或换行符。中括号可以组合其他特殊字符使用，如`[ab]*`匹配任意多的a或b。

# 基础语法

省略版

替换：

- `()`：把搜索中的某一部分用括号括住，让它可以用在替换里
- `$1`：替换文本中，`$n`代表第几个括号中的内容。`n`从1开始.

如：

- 文本：`"www.baidu.com; www.something.cn;"`
- 搜索：`www\.(.*?)\.(.*?)`;
- 替换：`abc,$1,$2`;
- 结果：`"abc,baidu,com abc,something,cn"`

替换，以及上文没讲的其余语法不会涵盖在本次练习内。有兴趣的同学可以自己查找资料学习。

# 练习

现在你已经对算术的基本原理有了一定了解...

- 匹配一个任意字母，后面跟一个任意数字的组合
- 匹配一个一位或两位的数字。这个数字可以是更大的数字的一部分。
- 匹配所有开头为a的行。第一行保证为空行。（提示：使用\n作为开头结尾
- 匹配一个简单的网址。一个简单的网址由http://或https://开头，中间为a.b.c的格式，可以有一个/作为结尾。例如，"http://www.baidu.com"或"https://a.b.c/"。
- 匹配所有以public abstract开头的函数。需要匹配大括号内的完整函数体。（提示：用[\w\n\d ]而非.匹配函数体内的内容。后者不会匹配换行符和空格。

我没有准备样例（偷懒了），做了的同学可以自己造两个简单的例子试试

遇到问题请直接提问。以及，我不会检查练习。。。不想做也没关系。

# 一些编程的本质

我自己说的，仅供参考

上文提到，鲁迅没有说过：编程就是把问题不断分解为小问题，小问题分解为更小的问题，最终分解到已有的功能上去。

这句话里的“问题”是一开始所讲的——针对**数据**的计算和**状态**的控制。

把输入的数据经过你定义的过程处理成你想要的数据。比如说，计算一系列数的最大值，或者把字符串里的单词全部转为大写。

或者，根据输入的数据，经过你定义的过程去做一些事情。比如说控制机器人，从网上下载指定文件。

编程会让你学会如何定义过程，也就是教计算机做事。但是“输入的数据”和“你想要的数据”，即问题本身，比定义过程更重要。

# 如何学习

a.k.a. 我们学编程的时候是在学什么？

怎么学会教计算机做事呢？

写代码的能力 = 描述问题 + 知道问题如何分解 + 知道已有的功能 + 知道如何组合已有的功能

非常不负责任的拿去年的lab题目举个例子：

给定 $p, q, a_0 \dots a_n$ , 求解 $\int_q^p a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ 。

现在的问题：java里没有算积分的函数。

所以想到，如果是在数学卷子上看到，步骤应该是这样：

$\int a_0 + a_1x + a_2x^2 + \dots + a_nx^n = a_0x + \frac{a_1x^2}{2} + \dots + C$ , 然后代数，做减法。

# 如何学习

a.k.a. 我们学编程的时候是在学什么？

现在的问题是，如何计算  $a_0x + \frac{a_1x^2}{2} + \dots + C$ ？（问题）

要计算上面的式子，需要：让程序知道  $n, a_0 \dots a_n$ ，计算  $x^i, a(x^i), a_0x + \frac{a_1x^i}{2} + \dots$ ，即算乘方，算加法乘法除法，算  $n$  个数的加法，再把结果打印出来。（小问题）

你们应该已经知道怎么输入输出了和算加减乘除了。（已有的功能）

循环结构应该也学过，用这个让计算机把算加法这件事干  $n$  遍。（还是已有的功能）

对于乘方，我们知道乘方是做  $n$  次乘法，所以可以像上面算  $n$  个数的加法那样自己写一个乘方的函数（功能），也可以**进行一个搜索引擎的使用**，发现有一个函数 `Math.pow()` 是算乘方的，直接利用前人写好的东西。

搜索引擎是必须要常用的。不像高中那些数学公式，编程里前人写好的东西太多了，没有人全记得住。大伙都是有需要的时候现查的。



# 如何学习

a.k.a. 我们学编程的时候是在学什么？

最后动用一下你的逻辑思维把这些东西全都组合起来（这一块就是 知道一些范式，然后多练！）：

- 读入所有的a, 以及p, q
- 记那个公式是 $g(x)$ , 问题的答案是 $g(p) - g(q)$ 
  - $g(x)$ 是n个数的求和, 要求和必须先知道数等于什么。用循环结构依次计算累加
    - 第i个数是 $\frac{a_i x^i}{i+1}$ , 计算之
    - 把第i个数和前面的累加结果加起来
- 输出结果

```
// 伪代码
int a = ..., p = ..., q = ... // a = [1, 2, ...], p = 114, q = 514 读入环节
// g(p), g(q)
int sumP = 0, sumQ = 0;
for (int i = 0; i < a.length; i++) { // 循环, 你们现在只需要知道这个里面的两行执行n次, 第几次由i表示
    sumP = sumP + (a[i] * math.pow(p, i)) / (i + 1);
    sumQ = sumQ + (a[i] * math.pow(q, i)) / (i + 1);
}
print(sumP - sumQ) // 输出
```

# 其他问题

开始碎碎念

First of all...

我不是寄系/电子系的，可以混过去吗？

可以。但是编程以后是你吃饭的家伙之一。你可以把高等数学忘光光，但进了实验室多少还得写点代码。

# 如何让自己的代码看得懂

良好的代码风格

```
for(int i=0;i<a;i++){if(i< 3)
{
    x +=1;y+= z;
}}

if (a&&b&&c&&d&&e&&f) //bad
if (a && b && c && d && e && f)

for (int i = 0; i < totalLen; i++) {
    if (i < 3) {
        descOfX += 1;
        descOfY += descOfZ;
    }
}
```

显然上面的一坨难以阅读...

经常使用 `Ctrl + Alt + L` 格式化你的代码

使用有意义的变量名和方法名（project时会再讲，lab就还好）

写注释

# 我的代码在本地为什么报错了？

大佬这有一堆红字我...诶我看得懂

首先，虽然**报错信息**它是英文的，而且里面估计有114514个你没见过的词，但是它设计出来

是为了给人看懂的！！！！！！！！

main -> mian

列几个常见报错：

- 编译错误：少分号/变量名打错/类名或关键字打错/大括号错了
- 运行时错误：多了去了
  - `ArithmeticException: / by zero: "/by zero", as described`
  - `ArrayIndexOutOfBoundsException: Index 1 out of bounds for length 0`：数组越界
  - `NullPointerException: Cannot invoke "... " because ... is null`：空指针，下节课讲

# 本地跑代码没问题，为什么oj报错了？

很可惜，oj没错。

oj的工作机制：准备好多种不同的输入数据，检查你程序的输出对不对

对于一组数据，如果你的输出和数据相符，那这个数据就通过了（绿色Accepted）

题目给你的样例一般是较简单的一组，里面的数据相对更为复杂，更能检测你代码的正确性

错了就自己造样例或者通读代码好好检查吧QAQ

OJ的各种错误类型：

- Compile Error(CE)：编译错误。检查：主类应该是Main，开头应该没有Package，程序没有语法错误
- Runtime Error(RE)：程序报错了。一般来说是数组越界，或者你有个数除以0了。
- Timeout(TLE)：程序超时了。java作业不考察程序的高效性，所以一般是你的程序里有死循环。

```
for (int i = n; i >= 0; i++)
```

# 怎么解决错误?

大佬不是你的网页浏览器

错误的处理方法：报错->大佬这里怎么改->大佬这里怎么改->大佬这里怎么改->大佬这里怎么改->...

然后不进脑子，下次还错，然后**反复**错一些简单的问题，折磨自己也折磨sa最后成绩也不好看。

正确的处理方法：报错->**为什么错?**

- 报的这个错没见过->它是什么? ->学会->以后不犯错，或者学会解决相似问题的方法
- 没报错，但是oj说你错了/你发现结果不对：利用debug工具或者打印查看中间结果->找到逻辑漏洞->结束

# 怎么问问题？

什么问题合适？

虽然自主解决问题的能力必须有，但是提问也是必须的！SA，TA和教授们很喜欢你们提问！

SA不喜欢也得喜欢，人每个月是领工资的，不用担心麻烦人

提问的智慧：<https://github.com/tvvocold/How-To-Ask-Questions-The-Smart-Way>

建议所有人回去通读一遍。我们和sa作为助教肯定不会像文章里那么的苛刻，可以按这个省流版本来：

- 以下所有的都是态度问题。不要把其他人当成保姆：(
- 问之前先用搜索引擎查一下，SA不是你的浏览器（推荐使用Google，其次bing，百度狗都不用）
- 问之前简单翻翻聊天记录。
- 问之前先根据你的代码和报错自己解决一下问题。如果没有成功，问的时候附上你的解决方法。
- 问的时候提供报错和代码。如果无法提供代码（怕查重），说思路，或者小窗问sa。不提供这些内容的问题就像“我今年高考数学倒数第二道大题错了，为什么？”
- 问的时候带上你的脑子。你得到的会是思路和解决方法，而非具体的代码。
- 问之前方便的话自己试一下。形如“a这样用会不会报错”这种问题，很多时候是自己敲进去跑一下就知道的事

# 老师没教怎么办？

啥都不会，一看就是上大学上的（不是玩原神玩的）

## 自己学。

大学，至少是计算机课程，没有超纲。

别人写作业写仨小时，你用老师没教的牛逼方法三分钟把作业秒了，那你牛逼，活该多玩三小时。

（老师教的你也得会，不过编程会了后面的前面的也应该会了。）

而且，知识太多了，教授的课程时间有限，不可能什么都讲。

讲计算机的书似乎没有其他学科那么成体系，而且越到后面越少（虽然足够cover计算机系课程了），到最后你只能去读官方文档，甚至源代码和论文。

b站和ytb上不少很好的课，大伙可以跟着学学。



# 如何使用搜索引擎

远离csdn

搜索引擎排名：谷歌大于bing，除非你在查很基础很基础的东西，否则不要百度。

问搜索引擎的方法：

- 问ArrayList是什么：java arraylist
- 问java怎么算乘方：java 乘方 / java calculate power

少点转折和语气词，只用关键词

推荐使用谷歌，或者bing的国际版，然后用英文关键词搜索。这样你能搜到StackOverflow的答案，能解决99%的问题。

中文社区真的不行，少看csdn。StackOverflow上的回答一般会仔细分析问题的原理，给出多种解法，还有很多人的讨论。csdn上一般只贴莫名其妙的代码，还不一定好使。（对于java这门课来说...勉强够用）

btw，科学上网几乎是必备的。你需要流畅的访问Google，StackOverflow，Github，Chatgpt，各种乱七八糟的官网，以及使用各种包管理器。

# 如何看待LLM?

天坑

个人不建议在第一门编程课上使用ChatGPT。一定不要用它写代码，也很不建议问它问题。

- 写代码是需要大量练习的。用一学期chatgpt等于高中抄一学期数学作业，到头啥也做不出来。
- 使用搜索引擎和向别人提问是很有用的技能。有的东西chatgpt没法告诉你，前两者可以，但是需要一点点练习。
- 它不总是对。可能你写代码也就半小时，但是问chatgpt五分钟，debug两小时。
- 而且你现在没有判断他对不对的能力。
- LLM没有设计代码的能力。它可以很好的写一个函数，但遇到几千行规模的代码无能为力。
- 如果十个人同时用了chatgpt写一份作业，它给出的代码估计差不多。被查重挂科了会很难看。

这玩意有点像搜题软件。不过软件能保持正确率，它不能。而且用过搜题软件的大伙应该都明白，需要自制力...

# 预告

- 类型系统，变量，内存
- 有时间就讲点控制语句