



大连理工大学
未来技术学院 / 人工智能学院
SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

人工智能程序设计

刘洋

理论课（全体成员）

4-11周，按照课表推进。

实践课（全体成员）

第6周开始，在学校机房上机。

书院强基会客室（自愿）

第6周开始，在未来书院活动室由助教进行辅导。

拔尖训练营/“三早”科学家团队（自愿）

课程开始后进行选拔（待大家完成学院发布的程序设计调查问卷后确定方案），后期进行项目制培养。



大连理工大学

未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

程序设计&人工智能

计算机

计算机语言

程序设计

人工智能



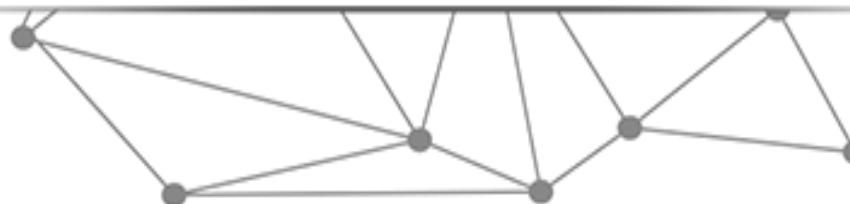
大连理工大学

未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

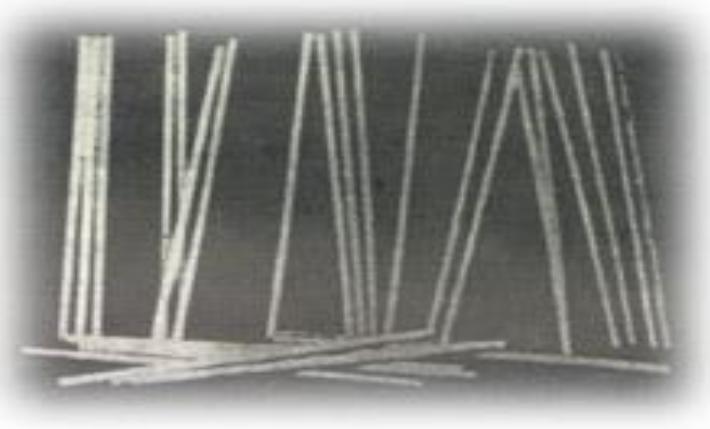
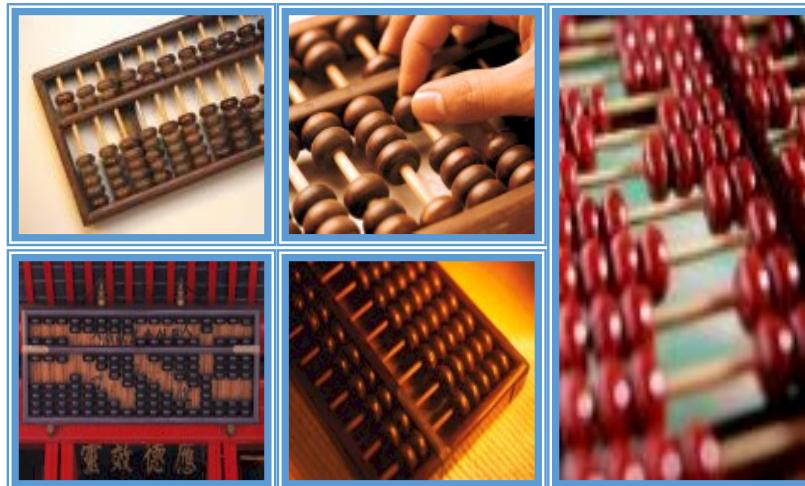


计算机的概念



计算机的起源

古人曰：“运筹于帷幄之中，决胜于千里之外。”筹策又叫**算筹**，它是中国古代普遍采用的一种计算工具。公元前500年，中国南北朝时期的数学家**祖冲之**，借用算筹作为计算工具，成功地将圆周率计算到小数点后的第七位。

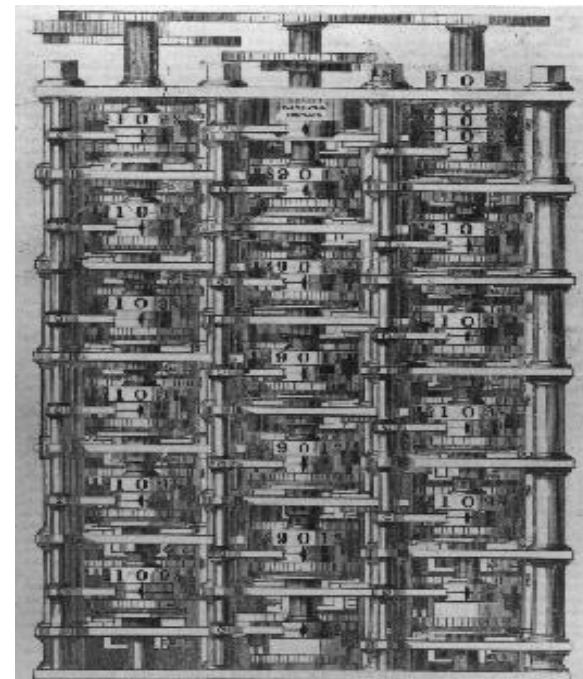


中国古代发明的**珠算盘**，直到今天，它仍然是许多人钟爱的“计算机”。对世界数学的发展产生了重要的影响。

机械计算机

机械计算机由法国的帕斯卡 (B. Pascal) 发明出来。帕斯卡设计的计算机是由一系列齿轮组成而用发条做为动力的装置，这种机器只能够做6位加法和减法。然而，即使只做加法也有个“逢十进一”的进位问题。

这称为“人类有史以来第一台计算机”，后来人们为了纪念它将一种计算机的高级语言命名为“PASCAL”。



现代计算机的诞生

现代计算机孕育于英国、诞生于美国、并成长遍布于全世界。所谓“现代”是指利用先进的电子技术代替机械或机电技术。现代计算机经历了60多年的发展（从1945年至今），其中最重要的代表人物是英国科学家**艾兰·图灵**（A. M. Turing）和美籍匈牙利科学家**冯·诺依曼**（Von Neumann），他们为现代计算机科学奠定了基础。

图灵和图灵机



图灵对现代计算机的主要贡献有两个：

- (1) 建立图灵机 (Turing machine) 理论模型；
- (2) 提出定义机器智能的图灵测试 (Turing test)。

1936年图灵发表了一篇论文：《论可计算的数及其在密码问题的应用》，首次提出逻辑机的通用模型。现在人们就把这个模型机称为图灵机，缩写为TM。TM由一个处理器P、一个读写头W/P和一条存储带M组成。

1950年他发表了另一篇著名论文：《计算机器与智能》。指出如果一台机器对于质问的响应与人类做出的响应完全无法区别，那么这台机器就具有智能。这一论断称为图灵测试，它奠定了人工智能理论的基础。

为纪念图灵的理论成就，美国计算机协会（ACM）专门设立了图灵奖。从1966年至今已有多位各国第一流的计算机科学家获得此项殊荣，成为计算机学术界的最高成就奖。

现代计算机

ENIAC是电子数值积分计算机 (The Electronic Numerical Integrator and Computer) 的编写。

1943年，二次世界大战关键时期，美国因新式火炮设计需要运算速度更快的计算机，宾州大学莫尔学院莫奇莱教授和他的学生埃克特博士用军方提供的48万美元设计完成.



冯·诺依曼介入ENIAC的工作是偶然的。冯·诺依曼到达莫尔学院计算机研究实验室看了研制中的计算机之后，提的第一个问题就是这台计算机的逻辑装置和结构，而这正是莫克利等人所谓判别真正的天才的标志。

冯·诺依曼对ENIAC机不足之处进行认真分析，并讨论全新的存储程序的通用计算机方案。当军方要求比ENIAC性能更好的计算机时，他便提出EDVAC（埃德瓦克）方案。

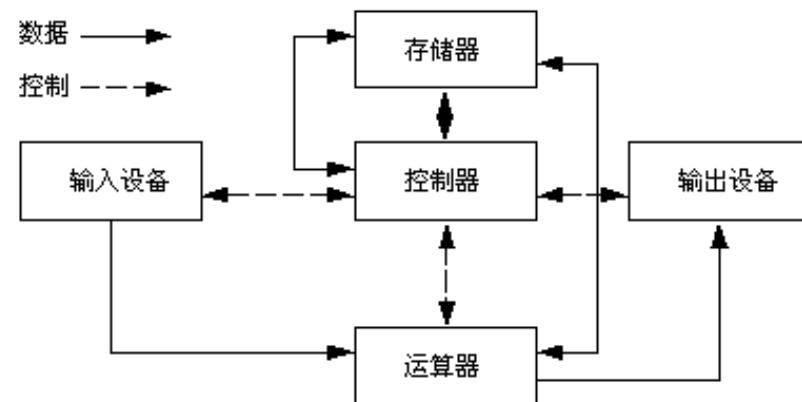


图 2.2 冯·诺依曼结构图

计算机的概念

- 计算机是根据指令操作数据的设备
- 具备功能性和可编程性两个基本特性

功能性能指对数据的操作，表现为数据计算、输入输出处理和结果存储等。

可编程性指它可以根据一系列指令自动地、可预测地、准确地完成操作者的意图

计算机的发展与摩尔定律

- 计算机硬件所依赖的集成电路规模按照摩尔定律以指数方式增长
- 计算机运行速度也接近几何级数快速增加
- 计算机所能高效支撑的功能不断丰富发展
- 程序设计语言经历了从机器语言到高级语言的发展过程，朝着更接近自然语言的方向发展。

- 第一阶段：1946年到1981年，“计算机系统结构阶段”
- 第二阶段：1982年到2007年，“计算机网络和视窗阶段”
- 第三阶段：2008年至今，“复杂信息系统阶段”。
- 第四阶段：约20年后未来的某个时期开始，“人工智能阶段”



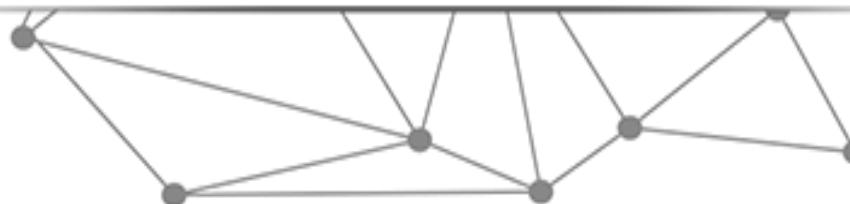
大连理工大学

未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT



计算机语言



日常生活中是常用的是十进制，但计算机中使用的是二进制，为了书写和口读方便，还采用了八进制、十六进制等。

在计算机中所有信息都必须以二进制形式进行编码。

1. 各种进位计数制及其表示法

进位计数制就是按进位方法进行计数。日常生活中人们已习惯于“逢十进一”的十进制计数，它的特点是：

(1) 用十个符号表示数。常用0、1、2、3、4、5、6、7、8、9符号，这些符号叫做数码。

(2) 数4024中的两个4表示不同的值，可写成多项式的形式：

$$4 \times 10^3 + 0 \times 10^2 + 2 \times 10^1 + 4 \times 10^0$$

上式中的 10^3 , 10^2 , 10^1 , 10^0 分别是个位、十位、百位、千位。这“个、十、百、千……”在数学上称为“权”。

数字表述与信息编码

(3) 十进制有0到9共十个数码，数码的个数称为基数。十进制的基数是10。

当计数时每一位计到十往上进一位，也就是“逢十进一”。所以基数就是两相邻数码中高位的权与低位权之比。

(4) 任一个十进制数N可表示为：

$$\begin{aligned} N &= \pm [a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \dots + a_0 \times 10^0 \\ &\quad + a_{-1} \times 10^{-1} + \dots + a_{-m} \times 10^{-m}] \\ &= \pm \sum a_i \times 10^i \end{aligned}$$

不难看出上式是一个多项式。式中的m、n是幂指数，均为正整数； a_i 称为系数，可以是0到9十个数码符号的任一个，由具体的数决定；10是基数。

数字表述与信息编码

对于二进制，数N可表示为：

$$N = \pm [a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \dots + a_0 \times 2^0 + a_{-1} \times 2^{-1} + \dots + a_{-m} \times 2^{-m}] = \pm \sum a_i \times R^i$$

基数是2，而数码符号只有0和1两个，进位为“逢二进一”。

对于八进制，数N可表示为：

$$N = \pm [a_{n-1} \times 8^{n-1} + a_{n-2} \times 8^{n-2} + \dots + a_0 \times 8^0 + a_{-1} \times 8^{-1} + \dots + a_{-m} \times 8^{-m}] = \pm \sum a_i \times R^i$$

基数是8，可用8个数码符号：0、1、2、3、4、5、6、7，进位为“逢八进一”。

对于十六进制，数N可表示为：

$$N = \pm [a_{n-1} \times 16^{n-1} + a_{n-2} \times 16^{n-2} + \dots + a_0 \times 16^0 + a_{-1} \times 16^{-1} + \dots + a_{-m} \times 16^{-m}] = \pm \sum a_i \times R^i$$

基数是16，可用16个数码符号：0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F进位为“逢十六进一”。

计算机与二进制数

计算机为什么要采用二进制呢？

- (1) 二进制数只有0、1两个状态，**易于实现**。例如电位的高、低，脉冲的有、无，指示灯的亮、暗，磁性方向的正反等等，都可以表示1、0。这种对立的两种状态区别鲜明，容易识别。
- (2) 二进制的**运算规则简单**。对于每一位来说每种运算只有四种规则。
- (3) 二进制信息的**存储和传输可靠**。两个稳态很容易识别和区分，所以工作可靠。
- (4) 二进制节省设备。从数学上推导，**采用** $R=e \approx 2.7$ **进位数制实现时最节省设备**，据此，采用三进制是最省设备的，其次是二进制。
- (5) 二进制可以用逻辑代数作为**逻辑分析与设计的工具**。逻辑代数是研究一个命题的真与假的数学工具，因此可以把二进制“0”和“1”作为真和假来看待。

计算机与二进制数



当然，二进制数也有它的缺点。

第一个缺点是人们不熟悉、不易懂，人们熟悉的是十进制。

第二个缺点是书写起来长，读起来不方便，为克服这个问题，又提出了八进制和十六进制。



在计算机中数的符号也用二进制数码表示，通常取正数的符号为0，负数的符号为1。

计算机中，涉及到小数点位置时，数有两种表示方法，即定点表示和浮点表示。

1) 定点数表示

任意一个二进制数总可以表示为纯整数（或纯小数）和一个 2 的整数次幂的乘积。例如，二进制数N可写成：

$$N = 2^P \times S$$

其中，S称为N的尾数；P称为N的阶码；2称为阶码的底。尾数S表示了N的全部有效数字，阶码P指明了小数点的位置。此处P、S都是用二进制表示的数。

当阶码为固定值时，称这种表示法为数的定点表示法。这样的数称为定点数。

计算机中定点与浮点数表示



大连理工大学

未来技术学院 / 人工智能学院
SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

如 $P=0$, 且尾数 S 为纯整数, 这时定点数只能表示整数, 称为**定点整数**。

如 $P=0$, 且尾数 S 为纯小数, 这时定点数只能表示小数, 称为**定点小数**。

定点数的这两种表示法, 在计算机中均有采用。究竟采用哪种方法, 都是事先约定的。

在计算机中数的符号也用二进制数码表示, 通常取**正数的符号为0**, **负数的符号为1**, 在机器内定点数中小数点的规定:

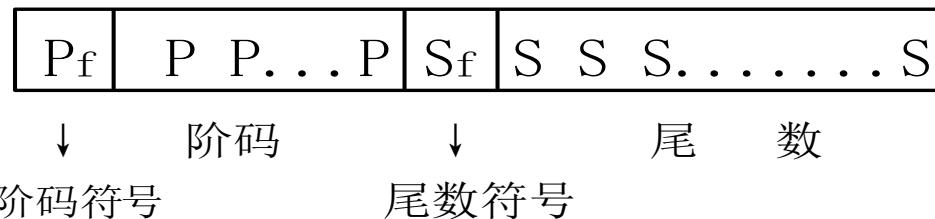
定点小数——**约定小数点在符号位与最高数值位之间**

定点整数——**约定小数点在最低有效位后面**

当定点数的位数确定以后, 定点数表示的范围也就确定了。如果一个数超过了这个范围, 这种现象称为溢出。

2) 浮点数表示

如果阶码可以取不同的数值，并与尾数一并表示，称这种表示法为数的浮点表示法。这样的数称为浮点数。这时：其中阶码P用整数表示，可为正数或负数。用一位二进制数Pf表示阶码的符号位，当Pf=0时，表示阶码为正数；当Pf=1时，表示阶码为负数。而尾数S一般为纯小数，用定点小数来表示，同样用Sf表示尾数的符号，Sf=0表示尾数为正数（也就是N为正）；Sf=1表示尾数为负数。



可见，在机器中表示一个浮点数，要分为阶码和尾数两个部分来表示。一般来说，阶码部分的位数决定了数的表示范围，而尾数部分的位数决定了数的精度。

计算机中的编码

计算机只能识别1和0，因此在计算机内表示的数字、字母、符号等都要以二进制数码的组合来代表，这就是二进制编码。根据不同的用途，有各种各样的编码方案，较常用的有ASCII码、BDC码、汉字编码等。

1. ASCII码

ASCII码即美国标准信息交换码，在计算机界得到了广泛使用。由国际标准组织（ISO）确定为国际标准字符编码。为了与国际标准接轨，我国根据它制定了国家标准，即GB1988。

ASCII码采用七位二进制位编码，共可表示128个字符，它包括26个英文字母大小写符号、数字、标点符号、专用符号及控制符号（如回车、换行），计算机中常以8位二进制即一个字节为单位表示，因此ASCII码最高位取0。而扩展ASCII码最高位为1，又可表示128个符号是一些制表符。ASCII码表见表2.4。

计算机中的编码



大连理工大学

未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

高位 低位	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	“	2	B	R	b	r
0011	EXT	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	‘	7	G	W	g	w
1000	BS	CAN	(8	H	X	h	x
1001	HT	EM)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	—	=	M]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL



计算机中的编码

2. BCD码

由于人们日常使用的是十进制，而机器内使用的是二进制，所以，需要将十进制表示成二进制码。

BCD码用十个不同的码：

0000, 0001, 0010, 0011, 0100,

0101, 0110, 0111, 1000, 1001,

表示0~9的十进制码，但它的每位是用二进制编码来表示的，因此称为二进制编码的十进制（**Binary Coded Decimal**）。

BCD码十分直观，可以很容易实现与十进制的转换。在商业上有它特殊的意义。

例：

(0010 1000 0101 1001 . 0111 0100) _{BCD}=2859.74

3. 汉字编码

汉字编码是一门涉及语言文字、计算机技术、统计数学、等多学科的学科。常见的汉字编码有： 国标码，大五码，Unicode, UTF-8等

1) 国标码

汉字是世界上是庞大的字符集。国家标准GB2312—80提供了中华人民共和国国家标准信息交换用汉字编码，简称国标码。该字符集的内容由三部分组成：

第一部分是各类符号、各类数字以及各种字母、包括英文、俄文、罗马字母、日文平假名与片假名、拼音符号和制表字符，共687个，第二部分为常用汉字，有3755个汉字，通常占常用汉字的90%左右，按拼音字母顺序排列，以便于查找。第三部分为二级常用汉字，有3008个，按部首顺序排列。

2) 机内码

该编码是指一个汉字被计算机系统内部处理和存储而使用的代码。国标码的表示方法（**2字节**，每个字节高位为**0**）和英文字符的**ASCII**码（西文的机内码，**1字节**表示，高位为**0**）在计算机内会产生冲突。为了保证中西文兼容，既允许西文机码存在，又允许国标码存在，就将国标码的每个字节的最高位（第**7位**）置“**1**”，来保证西文机内码和国标码在计算机内部的唯一性。因此，汉字操作系统将国标码的每个字节的最高位均置为**1**，标识为汉字机内码，简称汉字内码。二字节机内码如下所示：

1

国标码第一字节

1

国标码第二字节

计算机中的编码

3) 字形码

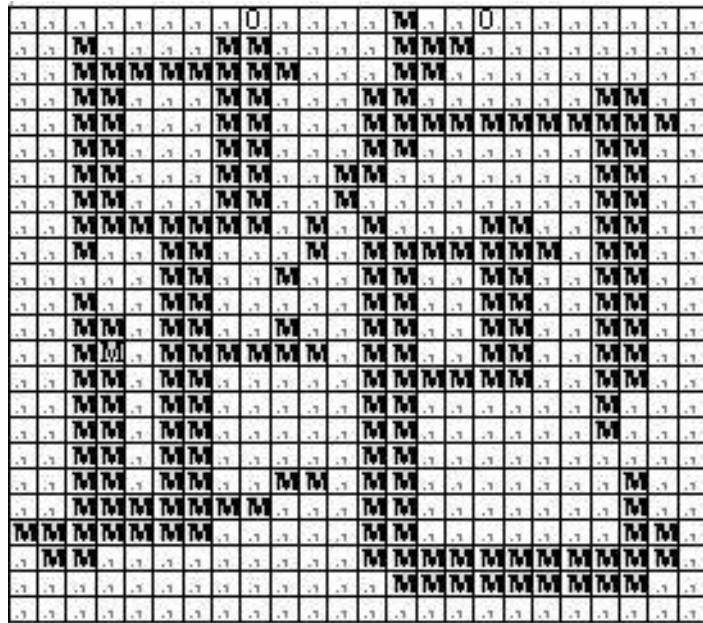
字形码就是描述汉字字形信息的编码，它主要分为两大类：字模编码和矢量编码。

字模编码是将汉字字形点阵进行编码，其方法是将汉字写在一个 24×24 的坐标纸上，在每个格子中就出现有墨和无墨两种情况，计算机就让每一个格子占一个二进制位，并规定有墨的地方用“1”表示，无墨的地方用“0”，然后将这些1、0按顺序排列下来，就成为汉字字模码。点阵的点越多时，表示（显示或打印）质量就也越高，也就越美观，但同时占用的容量也越大。点阵汉字表示简单，但在放大、缩小、变形后不够美观，

矢量汉字编码法。矢量汉字就是将汉字的形状、笔划、字根等用数学函数进行描述的方法。如TrueType就是一种，这样的字形信息便于缩放和变换，并且字形美观。

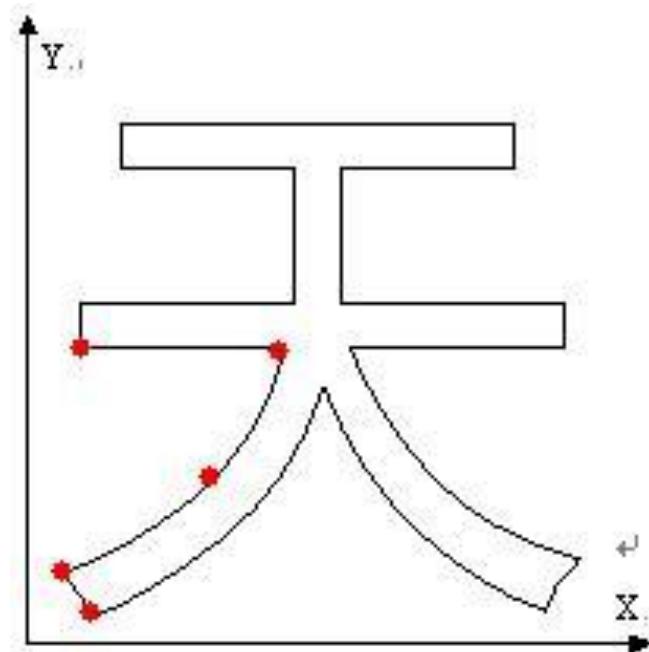
计算机中的编码

点阵字模



记录: 00 04 00
21 87 00
3F C6 00

矢量法



记录: 各点坐标和它们的关系

4) 汉字输入码

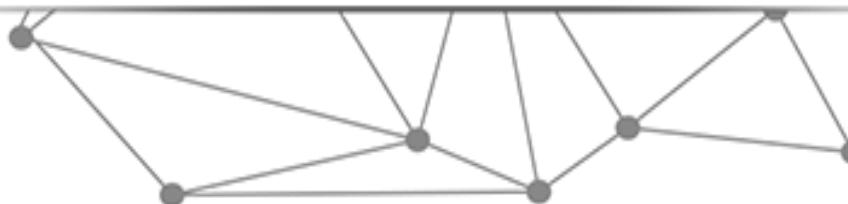
该编码指在键盘上利用数字、符号或字母将汉字以代码的形式输入。

由于存在多种输入编码方案，如区位码、首尾码、拼音码、简拼码、五笔字型码、电报码、郑码、笔形码等等，因此对常用的**6000**多个汉字和符号各有一套汉字输入码。

显然，一个汉字操作系统若支持几种汉字输入方式下，则在内部必须具备不同的汉字输入码与汉字国标码的对照表。这样，在系统支持的输入方式下，不论选定哪种汉字输入方式，每输入一个“汉字输入码”，便可根据对照表转换成唯一的汉字国标码。



程序设计

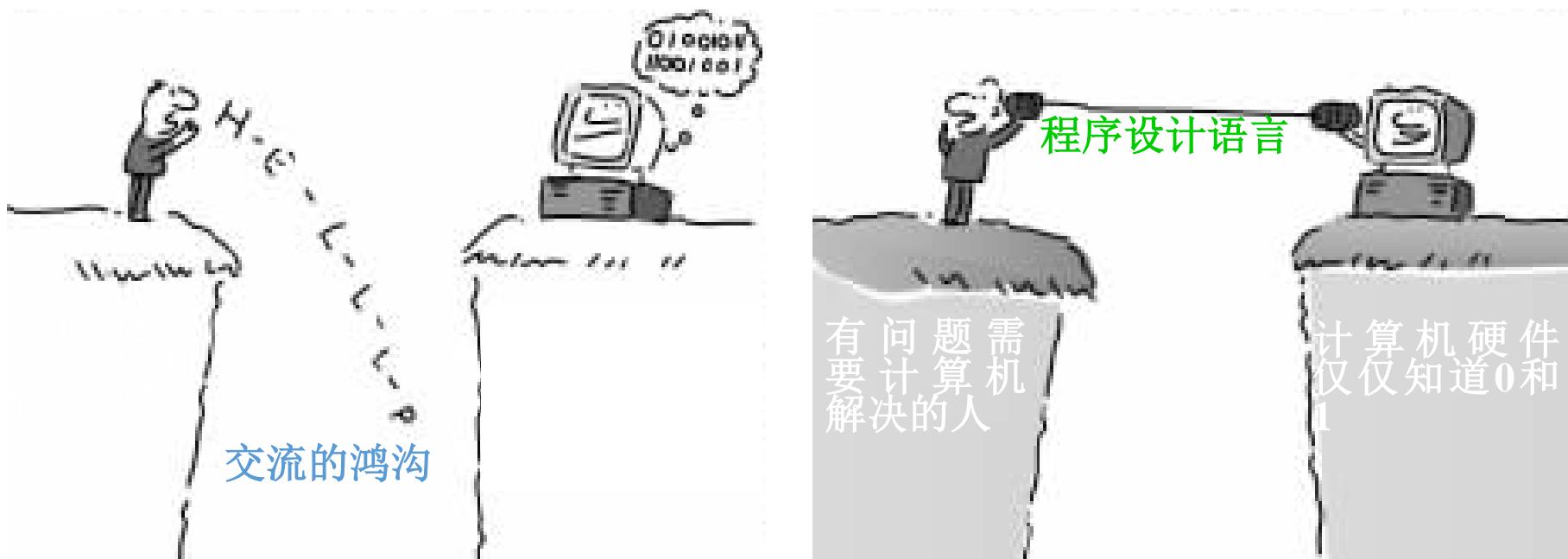


程序设计语言

为了让计算机解决一个实际问题，必须事先用计算机语言编制好程序。计算机语言使人们得以和计算机之间进行交流，其种类非常多，根据程序设计语言与计算机硬件的联系程度。

我们可以把它分为三类：

机器语言、汇编语言和高级语言。





计算机提供给用户的最原始的工具就是指令系统，我们用二进制编码的指令编写程序，然后输入计算机运行并得到预期的结果。以计算机所能理解和执行的“0”、“1”组成的二进制编码表示的指令，称为机器指令，或称为机器码。

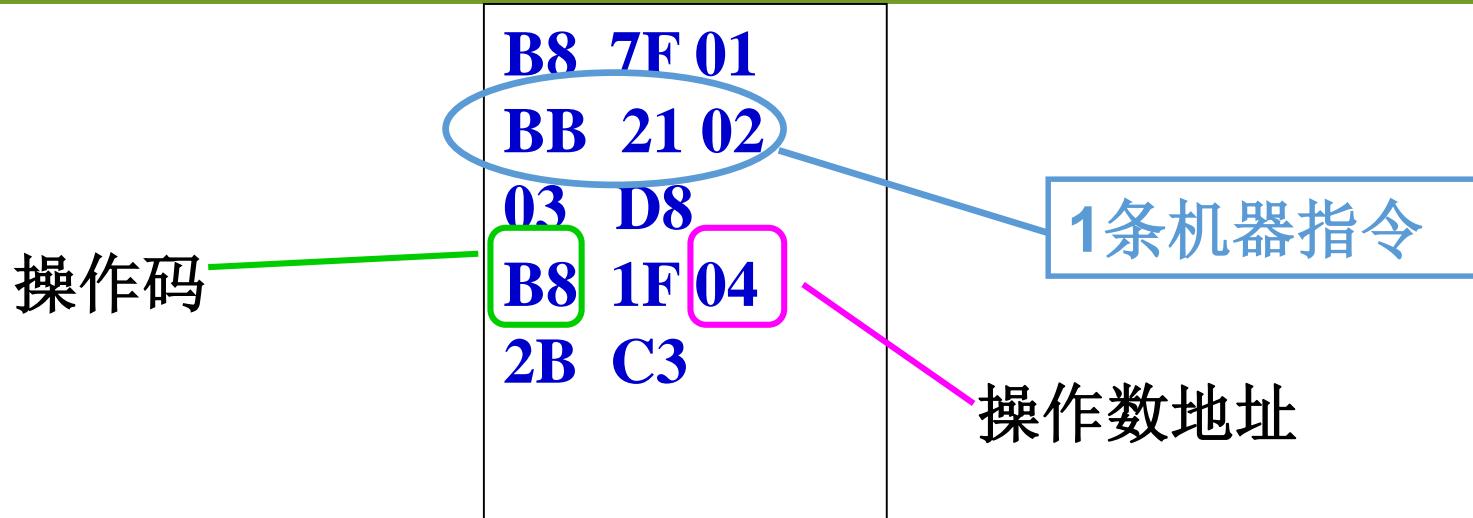
用机器指令编写的程序称为机器语言程序，或称为目标程序，这是计算机能够直接执行的程序。

机器指令的格式一般分为两个部分

操作码	操作数地址
-----	-------

应当注意：不同的机器，其指令系统是不同的，大多数现代计算机都设计了比较庞大的指令系统，以满足用户的需求。而且记不住、难理解、效率低、不易维护。

机器语言



(计算 $1055-(383+545)$ 的5条机器指令)

在MIPS计算机上求最大公约数的机器程序 (16进制表示)

```
27bdffd0 afbf0014 0c1002a8 00000000 0c1002a8 afa2001c 8fa4001c  
00401825 10820008 0064082a 10200003 00000000 10000002 00832023  
00641823 1483ffffa 0064082a 0c1002b2 00000000 8fbf0014 27bd0020  
03e00008 00001025
```

MISP计算机的每条机器指令均为32个二进位，用8个16进制数表示

汇编语言



大连理工大学

未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

为了克服机器语言的缺点，在科研人员的研究工作中很快就发明和产生了比较易于阅读和理解的汇编语言。所谓汇编语言，就是采用英文字母、符号来表示指令操作码、寄存器、数据和存储地址等，并在程序中用它们代替二进制编码数，这样编写出来的程序就称为符号语言程序或汇编语言程序。大多数情况下，一条汇编指令对应一条机器指令，少数对应几条机器指令。

汇编语言程序

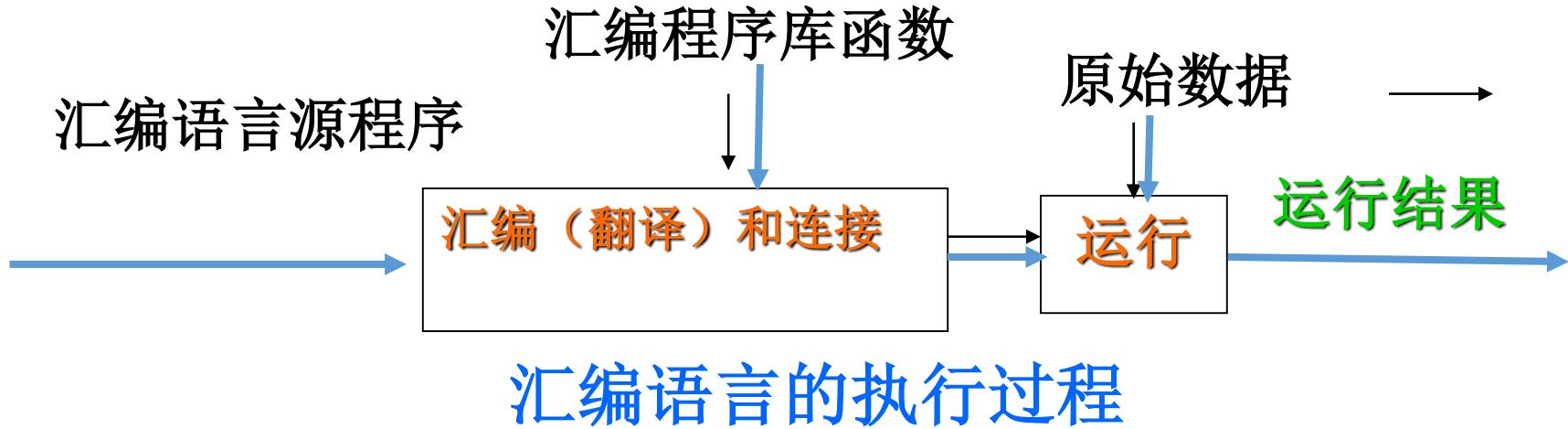
```
MOV AX 383  
MOV BX 545  
ADD BX AX  
MOV AX 1055  
SUB AX BX
```

对应的汇编语言程序

机器语言程序

B8	7F	01
BB	21	02
03	D8	
B8	1F	04
2B	C3	

(计算 $1055-(383+545)$ 的5条机器指令)



汇编语言的抽象层次很低，与机器语言一样，是与具体的机器密切相关的。针对某一种机器编写出来的程序，不能在其它机器上执行。即可移植性较差。一些复杂的运算通常要用一个子程序来实现，而不能用一个语句来解决，因此用汇编语言编写程序仍然相当麻烦。尽管如此，从机器语言到汇编语言，仍然是前进了一大步。这意味着人与计算机的硬件系统不必非得使用同一种语言。随着计算机程序设计技术的发展而出现的高级语言可以避免汇编语言的这些缺点。

高级语言

高级语言的出现是计算机编程语言的一大进步。它屏蔽了机器的细节，提高了语言的抽象层次，程序中可以采用具有一定含义的数据命名和容易理解的执行语句。这使得在书写程序时可以联系到程序所描述的具体事物，**比较接近人们习惯的自然语言**，是为一般人使用而设计的，处理问题采用与普通的数学语言及英语很接近的方式进行，并且不依赖于机器的结构和指令系统。如目前比较流行的语言有Python、Java、C/C++、Fortran、Pascal等。使用高级语言编写的程序通常能在不同型号的机器上使用，可移植性较好。

汇编语言程序

```
MOV AX 383  
MOV BX 545  
ADD BX AX  
MOV AX 1055  
SUB AX BX
```

$$S=1055-(383+545)$$



■ 程序设计语言包括编译执行和解释执行两种方式

程序设计语言是计算机能够理解和识别用户操作意图的一种交互体系，它按照特定规则组织计算机指令，使计算机能够自动进行各种运算处理。按照程序设计语言规则组织起来的一组计算机指令称为计算机程序。

■ 高级语言

高级语言区别于低级语言在于，高级语言是接近自然语言的一种计算机程序设计语言，更容易地描述计算问题并利用计算机解决计算问题。

例如：执行数字2和3加法的高级语言代码为：

```
result = 2 + 3
```

高级语言分类

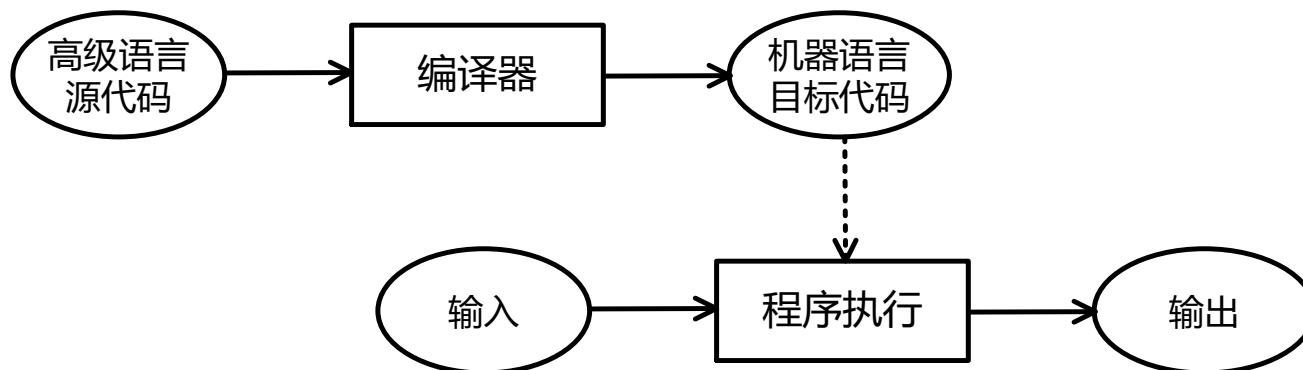
高级语言按照计算机执行方式的不同可分成两类

- 静态语言（编译）
- 脚本语言（解释）

这里所说的执行方式指计算机执行一个程序的过程，静态语言采用编译执行，脚本语言采用解释执行。

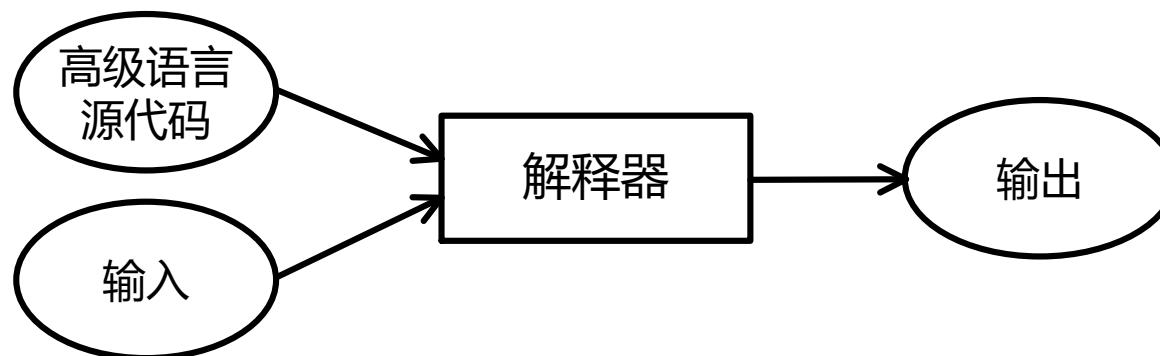
解释和编译

- 编译是将源代码转换成目标代码的过程，通常，源代码是高级语言代码，目标代码是机器语言代码，执行编译的计算机程序称为编译器



解释和编译

- 解释是将源代码逐条转换成目标代码同时逐条运行目标代码的过程。执行解释的计算机程序称为解释器。



→ 解释和编译

编译是一次性地翻译，一旦程序被编译，不再需要编译程序或者源代码。

- 对于相同源代码，编译所产生的目标代码执行速度更快。
- 目标代码不需要编译器就可以运行，在同类型操作系统上使用灵活。

解释则在每次程序运行时都需要解释器和源代码。

- 解释执行需要保留源代码，程序纠错和维护十分方便。
- 只要存在解释器，源代码可以在任何操作系统上运行，可移植性好

计算思维是区别于以数学为代表的逻辑思维和以物理为代表的实证思维的第三种思维模式。

编程是一个求解问题的过程

- 首先需要分析问题，抽象内容之间的交互关系
 - 设计利用计算机求解问题的确定性方法，
 - 进而通过编写和调试代码解决问题
- 这是从抽象问题到解决问题的完整过程。

计算机编程



程序

=

算法

+

数据结构



人工智能

Python



大连理工大学

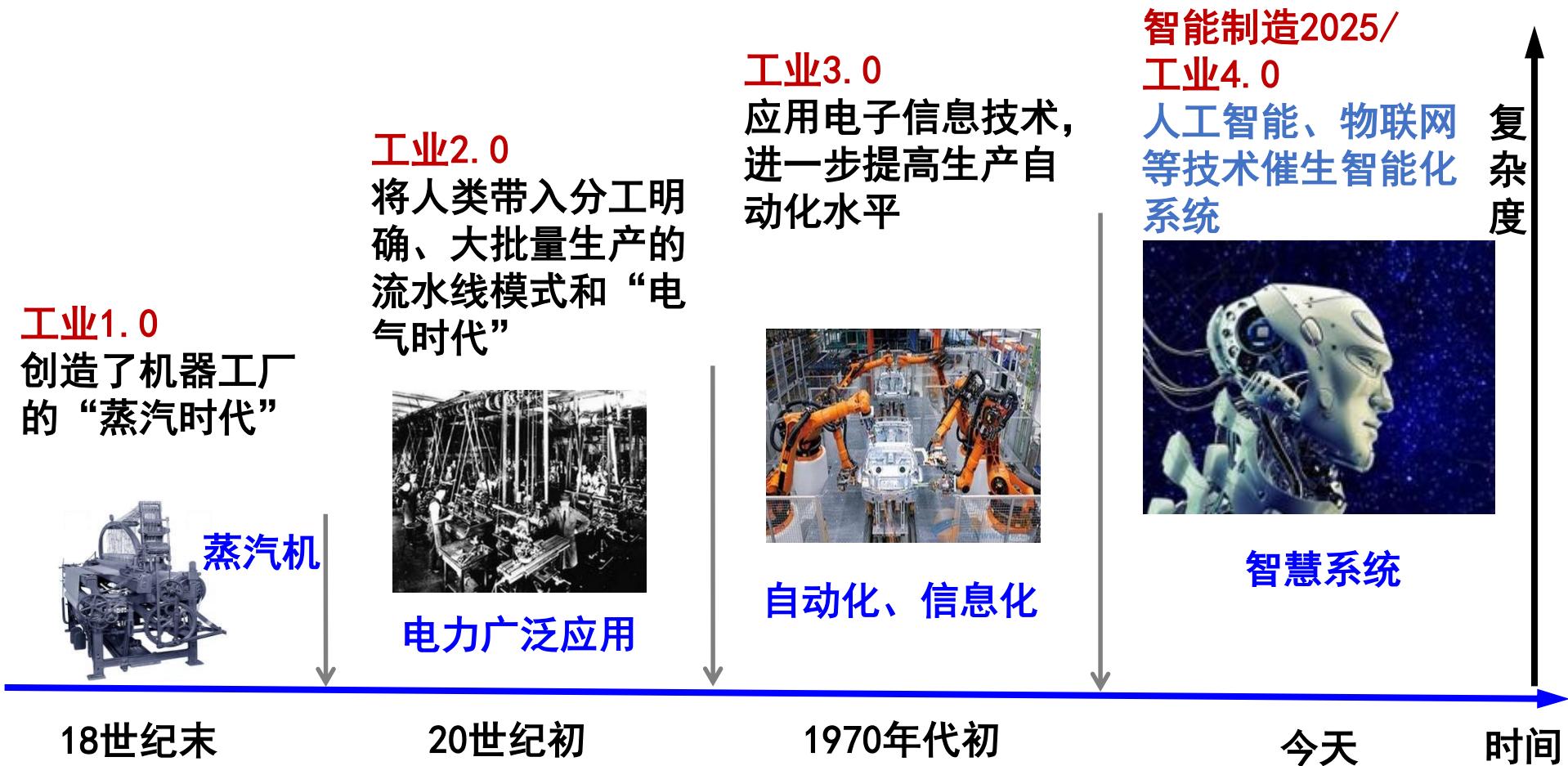
未来技术学院 / 人工智能学院

SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT



人工智能与 Python

智能化时代来临



什么是人工智能

- 人工智能（Artificial Intelligence），英文缩写为AI。它是研究、开发用于模拟、延伸和扩展人的智能的理论、方法、技术及应用系统的一门新的技术科学。



→ 人工智能发展史



大连理工大学
未来技术学院 / 人工智能学院
SCHOOL OF FUTURE TECHNOLOGY, SCHOOL OF ARTIFICIAL INTELLIGENCE, DUT

目标：人工智能

早期人工智能令世界兴奋



1950s 1960s 1970s 1980s 1990s 2000s 2010s Now

手段：机器学习
机器学习技术飞速发展



手中的AI



智能图像理解
智能美图



新闻推荐
智能搜索排序



智能出行
自动驾驶



智能会话
智能助理



识别:

语音、人脸、视频内容
应用: 安防、客服。。。

BI、商业流程自动化

互联网数据

购买转化、商品推荐、定
价、精准营营销、
社交媒体营销

机器翻译



券商股票:

智能+量化交易、智能投
顾、机器人理财

个性化推荐
智能物流
仓储机器人



银行、保险应用:

针对性电话营销、
贷款审批、信用卡欺诈

医学:

自动读片、辅助诊断、个
性化诊断、基因排序

教育:

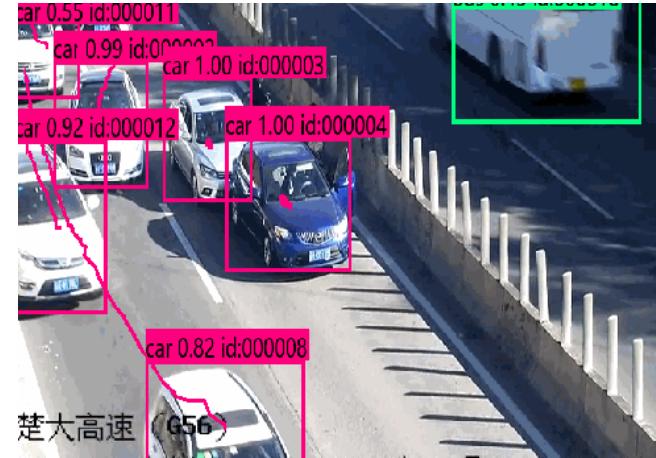
学习外语、智能选题

如何实现

目标：人工智能

手段：机器学习

实现：Python



Why Python?

- Python语言是一种解释型、面向对象、动态数据类型的高级程序设计语言
- Python语言是数据分析师的首选数据分析语言，也是智能硬件的首选语言

数据分析



创建复杂的Web应用
程序

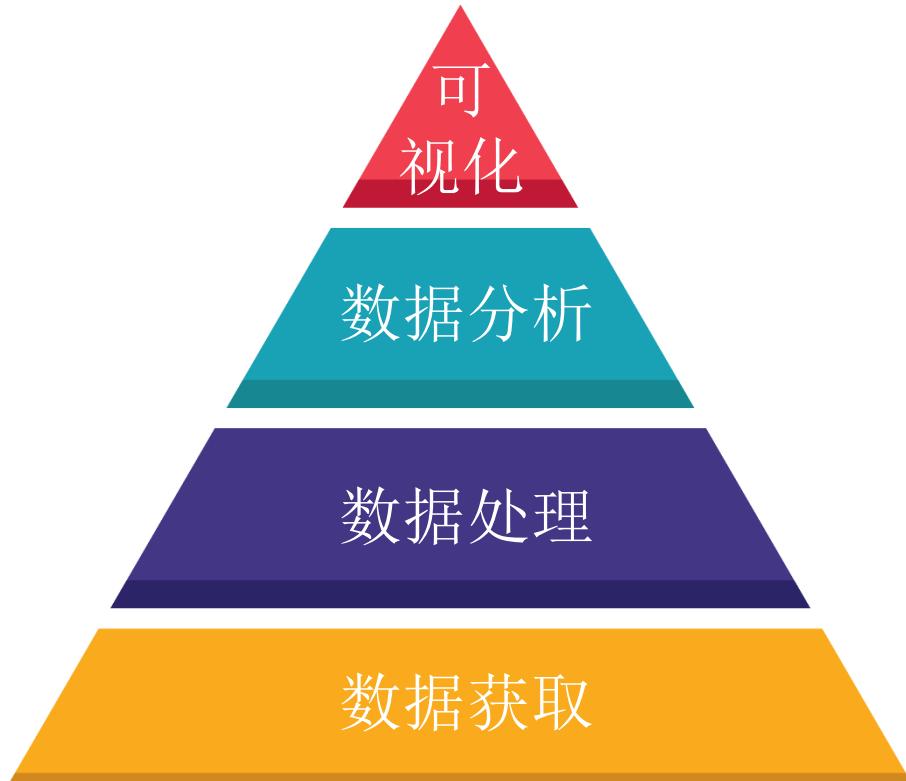
游戏开发

.....

网站与游戏开发

智能硬件开发

Why Python?



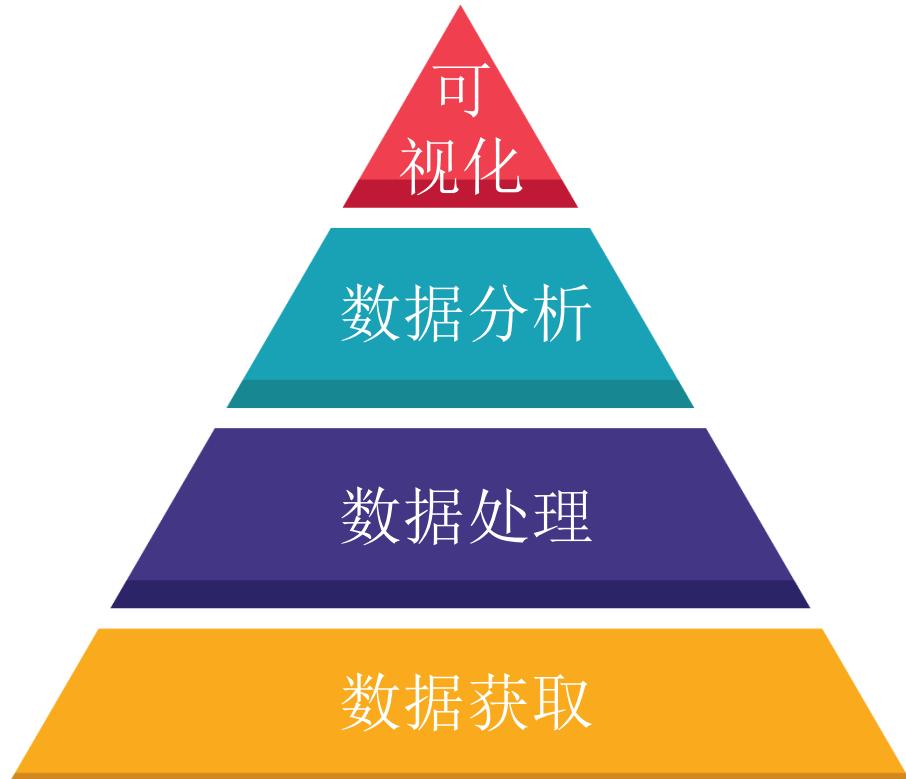
界面型: Excel、Powerpoint、PS.....
代码型: R、SAS、Python、Process.....

界面型: SPSS.....
代码型: R、SAS、Python.....

数据库: SQL, Hive.....
界面型: Excel、SPSS.....
代码型: VBA、Python、R、SAS.....

数据库: SQL, Hive.....
爬虫: Python、Java.....

Why Python?



界面型: Excel、Powerpoint、PS.....
代码型: R、SAS、**Python**、Process.....

界面型: SPSS.....
代码型: R、SAS、**Python**.....

数据库: SQL, Hive.....
界面型: Excel、SPSS.....
代码型: VBA、**Python**、R、SAS.....

数据库: SQL, Hive.....
爬虫: **Python**、Java.....

Python发展史

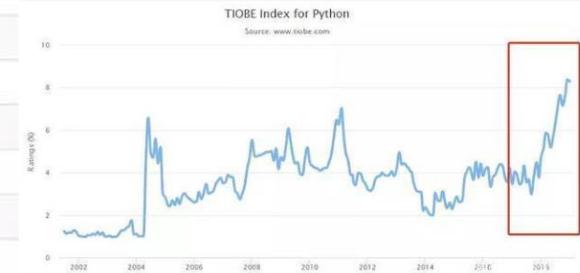
- Guido van Rossum 于1989年在荷兰国家数学和计算机科学研究所设计。
- Python 2.0版本于2000年10月发布。
- 在2008年12月， Python 3.0发布，此版本没有完全兼容之前的Python 2.0。
- Python也因此分为了Python 3.5派系和Python 2.7派系两大阵营。
- 2018年3月，该语言作者在邮件列表上宣布 Python 2.7将于2020年1月1日终止支持。



→ Python发展史

根据TIOBE排行榜的数据，Python获得了2018年年度编程语言

Year	Winner
2018	Python
2017	C
2016	Go
2015	Java
2014	JavaScript
2013	Transact-SQL
2012	Objective-C
2011	Objective-C
2010	Python
2009	Go
2008	C
2007	Python
2006	Ruby
2005	Java
2004	PHP
2003	C++



→ Python语言的特点



- 优雅、简单、明确（减少花哨、晦涩或以“炫技”为目的的代码）

```
#include<stdio.h>
main()
{
    printf("Hello World");
}
```

```
print "Hello World"
```

```
Hello World
```

C语言

Python语言

→ Python语言的特点

- 强大的标准库（“调包侠”）
- 良好的可扩展性
- 免费、开源

Python语言的特点

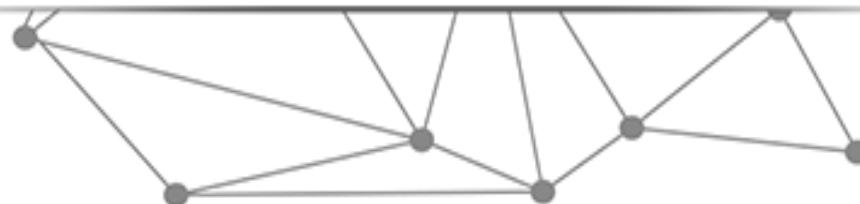
- 缺点一：运行速度慢
- 缺点二：加密难
- 缺点三：缩进规则
- 缺点四：多线程灾难

```
if True:  
    print "right"  
  
File "<ipython-input-1-ab7d6f176ce6>", line 2  
    print "right"  
        ^  
  
IndentationError: expected an indented block
```

```
if True:  
    print "right"  
  
right
```

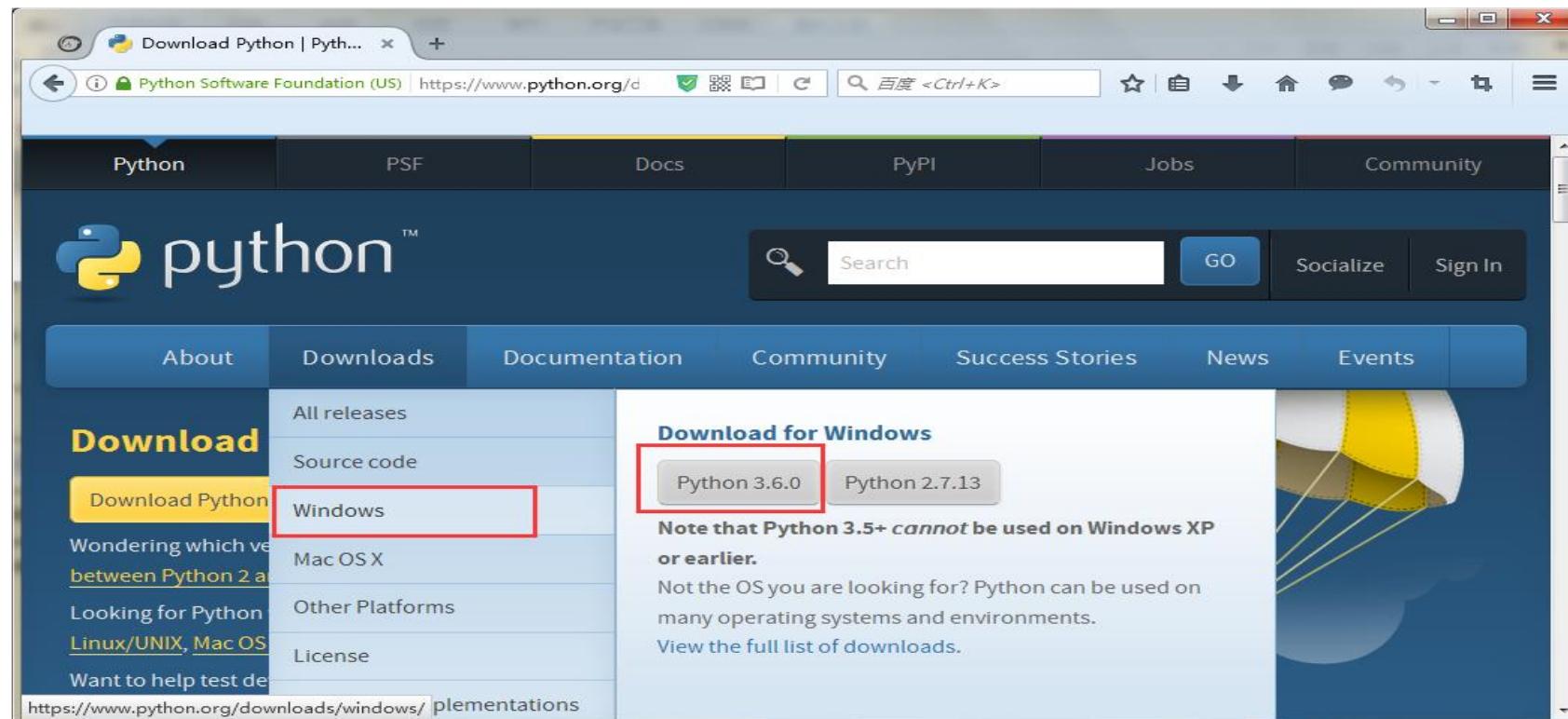


Python开发环境配置



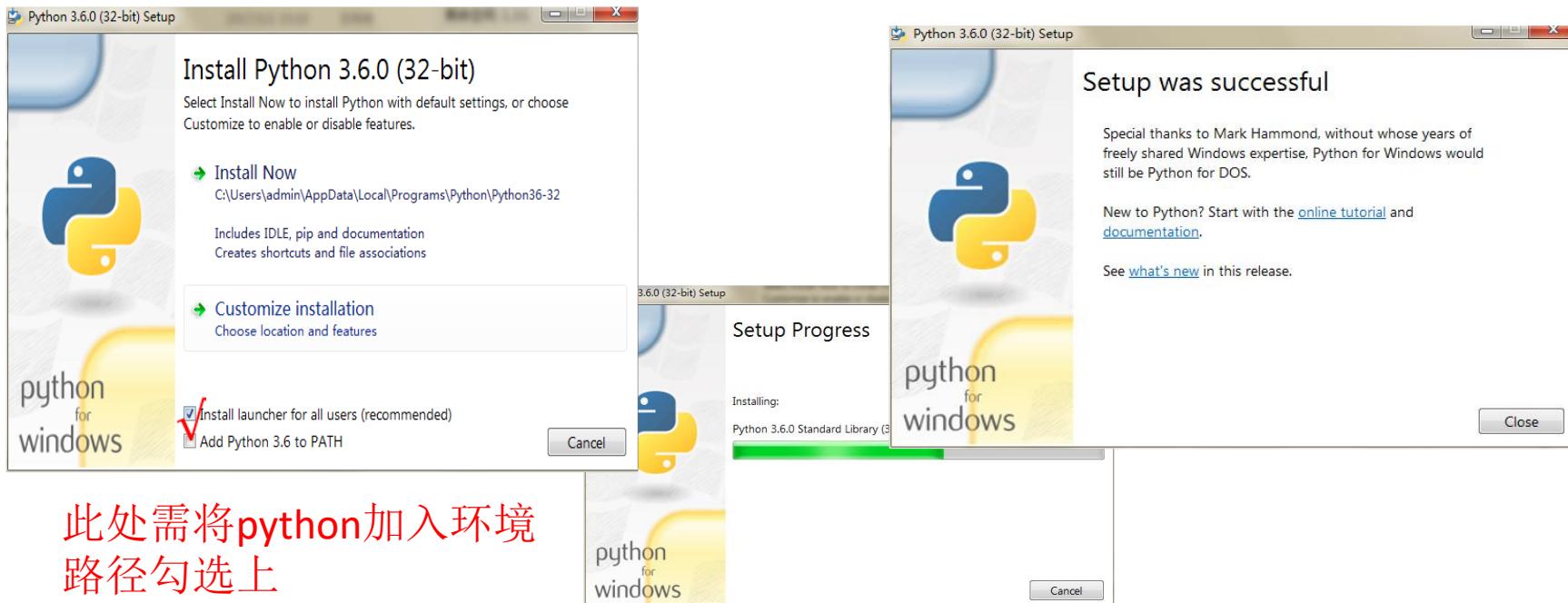
- 到Python主页下载并安装Python基本开发和运行环境，
网址：www.python.org/downloads/
- 根据操作系统不同选择不同版本
- 下载相应的Python 3.0系列版本程序

运行环境及安装



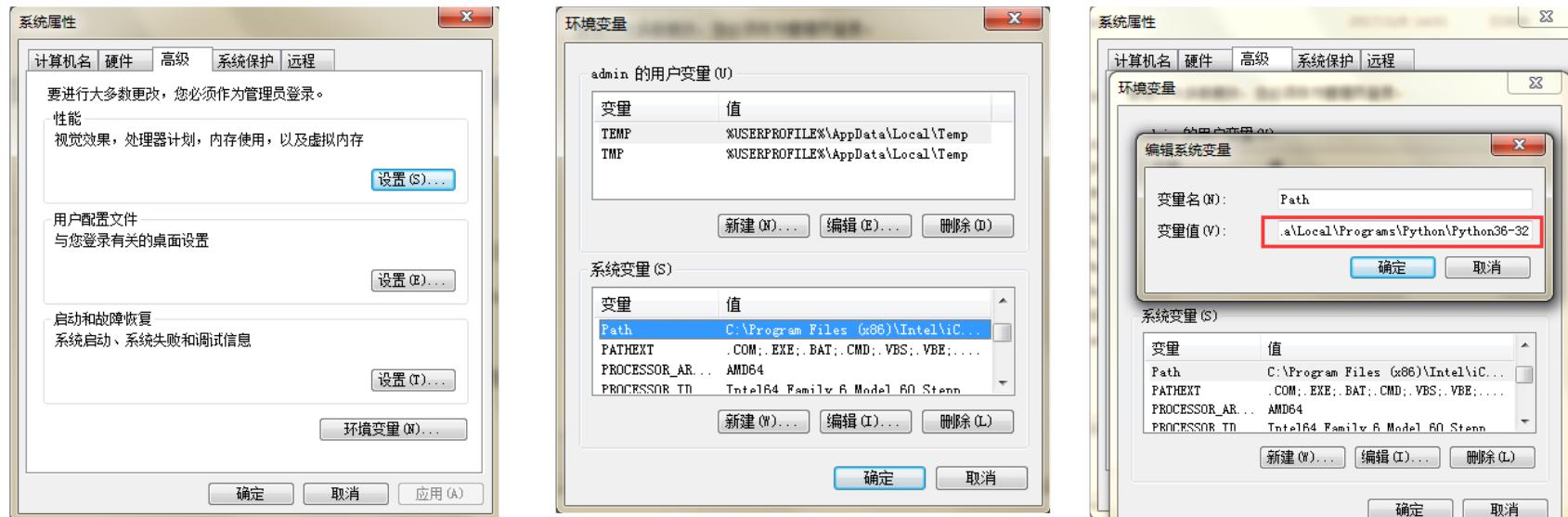
→ 运行环境及安装

(2) 下载并完成Python的安装。



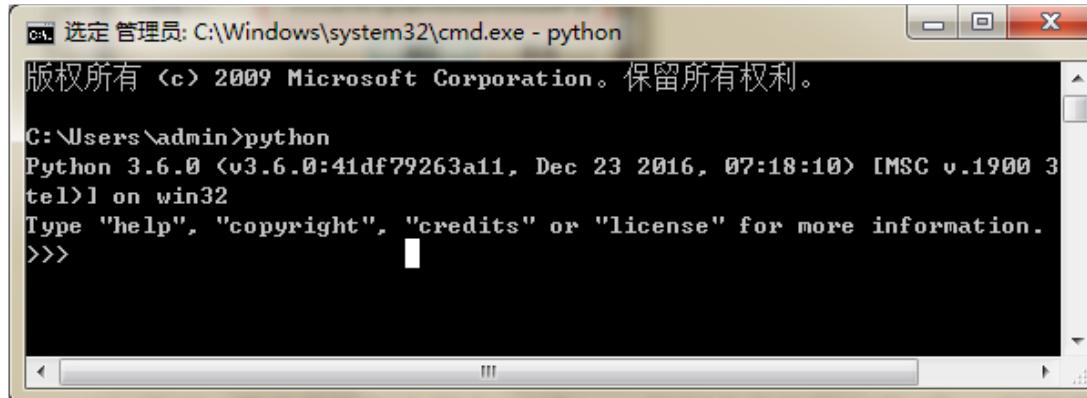
运行环境及安装

(3) 配置环境变量（如果上一步没有勾选添加环境变量则需要这一步，否则不需要）。



运行环境及安装

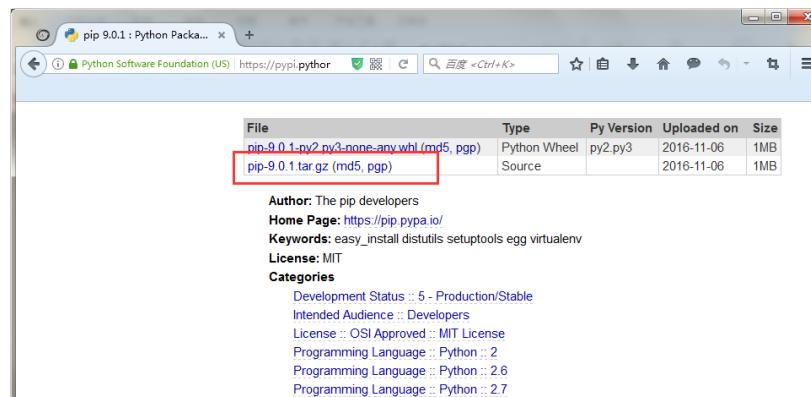
(4) 环境变量配置成功



```
选定 管理员: C:\Windows\system32\cmd.exe - python
版权所有 <c> 2009 Microsoft Corporation。保留所有权利。
C:\Users\admin>python
Python 3.6.0 (v3.6.0:41df79263a11, Dec 23 2016, 07:18:10) [MSC v.1900 3
tel>l on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

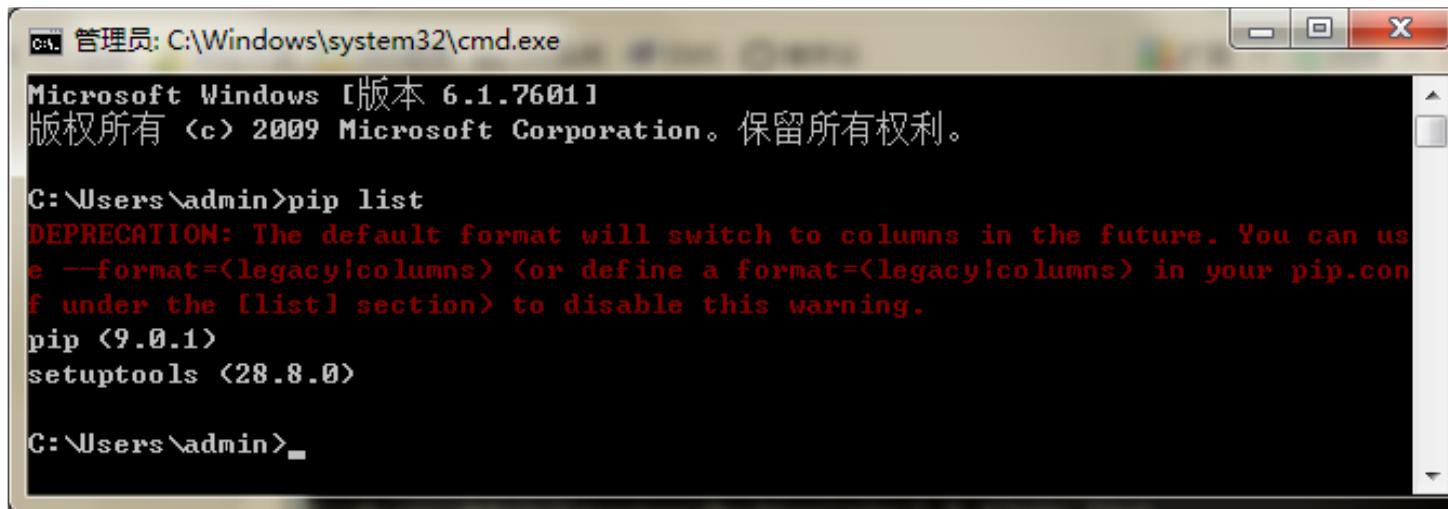
(5) 下载pip。下载网站:

[https://pypi.python.org/pypi/pip#downloads。](https://pypi.python.org/pypi/pip#downloads)



运行环境及安装

(6) 解压文件后，在Path最后面添加Scripts文件路径。再次打开控制台，输入pip list，出现下列信息表示pip安装成功



```
管理员: C:\Windows\system32\cmd.exe
Microsoft Windows [版本 6.1.7601]
版权所有 <c> 2009 Microsoft Corporation。保留所有权利。

C:\Users\admin>pip list
DEPRECATION: The default format will switch to columns in the future. You can use --format=<legacy|columns> (or define a format=<legacy|columns> in your pip.conf under the [list] section) to disable this warning.
pip <9.0.1>
setuptools <28.8.0>

C:\Users\admin>
```



运行

■ 方法1：启动Windows命令行工具，输入python

```
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.1600 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello World")
Hello World
>>>
```



■ 方法2：调用IDLE来启动Python图形化运行环境

The screenshot shows the IDLE Shell 3.11.5 interface. The title bar reads "IDLE Shell 3.11.5". The menu bar includes File, Edit, Shell, Debug, Options, Window, and Help. The main window displays the Python 3.11.5 interpreter output. It shows the Python version information, the copyright notice, and a simple "hello world" program run:

```
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> print("hello world")
hello world

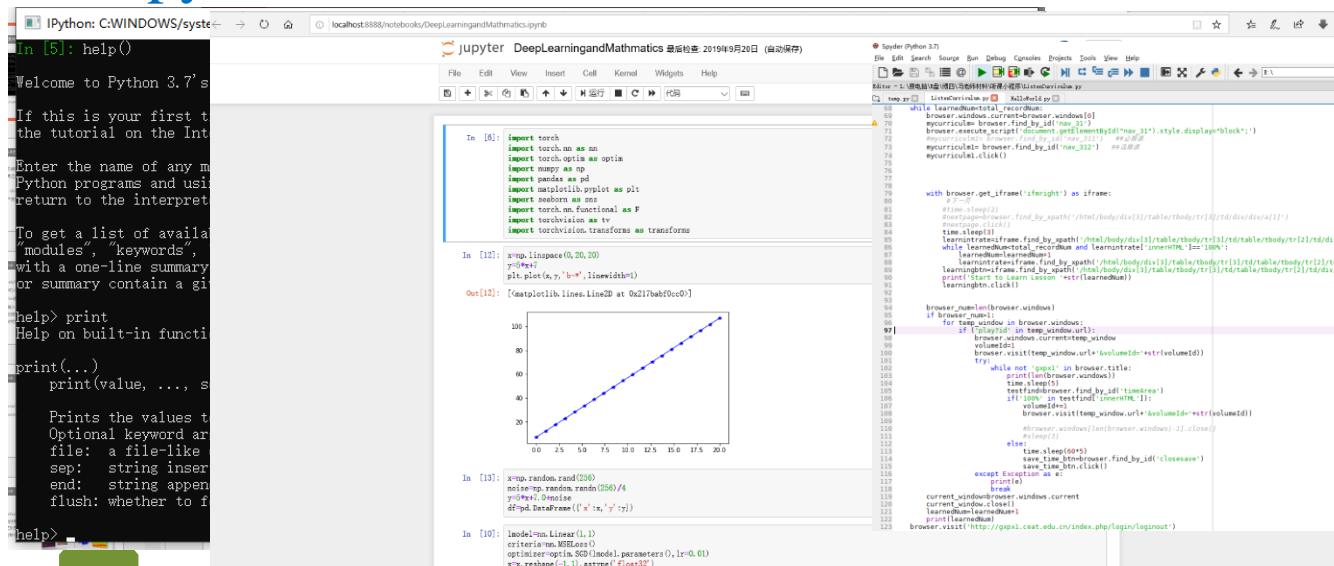
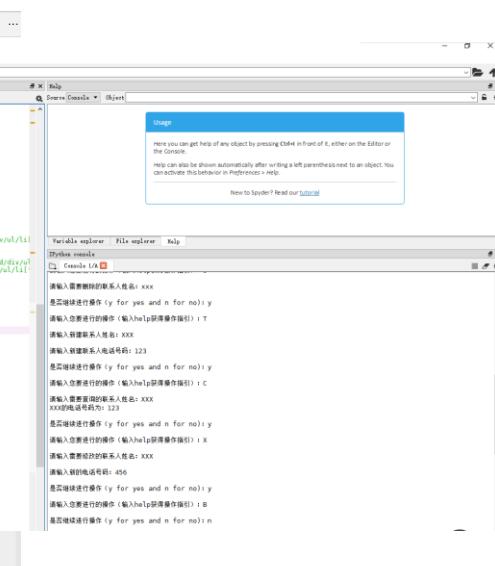
>>>
```

In the bottom right corner of the window, there is a status bar with "Ln: 5 Col: 0".

■ 方法3：将Python集成到VSCode、PyCharm等 面向较大规模项目开发的集成开发环境中

方法见QQ群

- Anaconda, 开源（Python）环境管理器
- Ipython
- Jupyter Notebook
- Spyder

```
In [5]: help()
Welcome to Python 3.7's

If this is your first time running this tutorial on the Internet, enter the name of any module, keyword or summary and use "return" to return to the interpreter.

To get a list of available modules, keywords, with a one-line summary or summary contain a git

help> print
Help on built-in function print(...)

Prints the values to file: a file-like object
sep: string inserted between values
end: string appended at the end
flush: whether to flush immediately

help>
In [6]: import torch
import torch.nn as nn
import torch.optim as optim
import numpy as np
import pandas as pd
import torch.nn.functional as F
import torchvision as tv
import torchvision.transforms as transforms

In [7]: x=np.linspace(0,20,20)
y=x*2
plt.plot(x,y,'b*-')
plt.show()

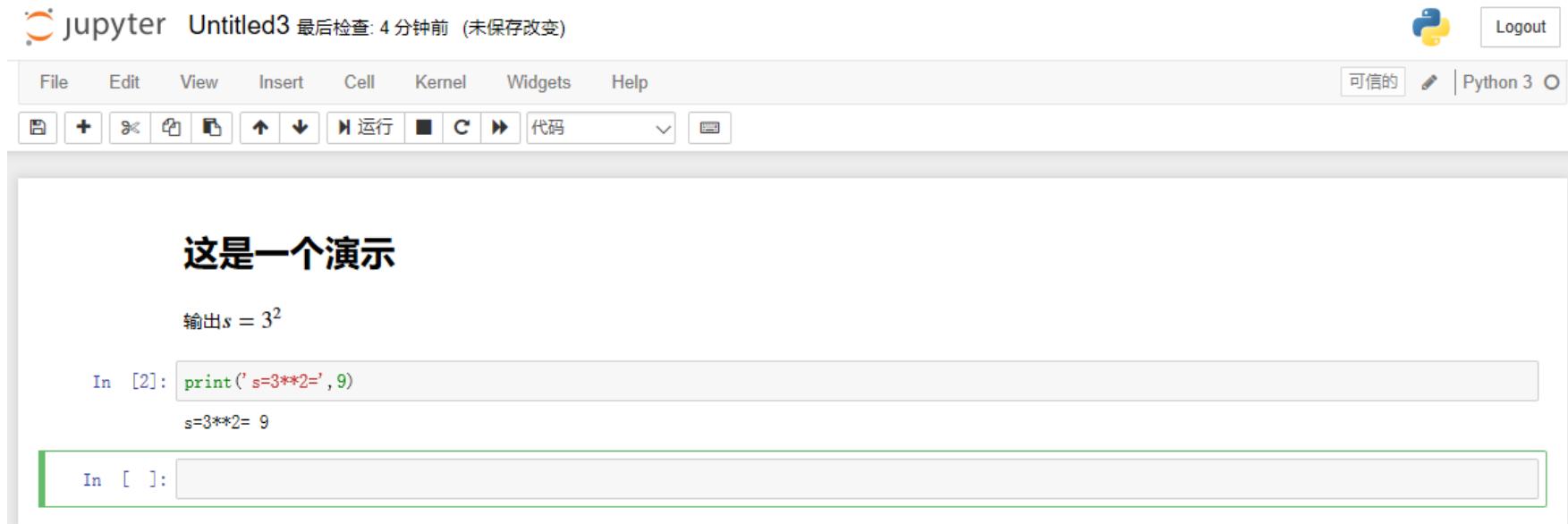
Out[7]: [Figure: A scatter plot showing a linear relationship between x and y. The x-axis ranges from 0.0 to 20.0, and the y-axis ranges from 0 to 100. The data points form a straight line starting at (0,0) and ending at (20,40).]

In [8]: model=Linear(1,1)
criterion=MSELoss()
optimizer=optim.SGD(model.parameters(),lr=0.01)
x=x.reshape(-1,1).numpy().astype('float32')

In [9]: for i in range(1000):
    optimizer.zero_grad()
    current_out=model(x)
    loss=criterion(current_out,y)
    loss.backward()
    optimizer.step()

In [10]: model.state_dict()
Out[10]: OrderedDict([('linear.weight', tensor([[ 0.0100]]), 'linear.bias', tensor([ 0.0000])])
```

- Julia+Python+R = Jupyter
- 基于Web技术的交互式计算文档格式

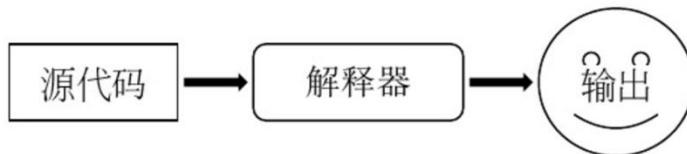




Python 输入输出

解释型语言Python

- Python语言是一种**解释型**、面向对象、动态数据类型的高级程序设计语言



解释型语言的执行方式



编译型语言的执行方式

编译型	解释型	混合型
c	JavaScript	JAVA
c++	Python	C#
GO	Ruby	
Swift	PHP	
Object-C	Perl	
Pascal	Erlang	

输入与输出

- Python输出方式有两种：

(1) 表达式语句

`str()`: 函数返回一个用户易读的表达形式。

`repr()`: 产生一个解释器易读的表达形式。

(2) `print()` 函数

In [8]: `output=' 欢迎学习AI程序设计'`

In [9]: `output`

Out[9]: ' 欢迎学习AI程序设计'

In [10]: `str(output)`

Out[10]: ' 欢迎学习AI程序设计'

In [11]: `repr(output)`

Out[11]: " 欢迎学习AI程序设计 "

In [12]: `print(output)`

欢迎学习AI程序设计

输入与输出

- Python输出方式有两种：

(2) print() 函数

In [8]: `output='欢迎学习AI程序设计'`

In [14]: `print(output)`

欢迎学习AI程序设计

```
className='AI程序设计'  
Time=6  
duaration=3
```

```
print(className, '在', Time, '点开始, 时长', duaration, '学时')
```

AI程序设计 在 6 点开始，时长 3 学时

```
print ('%s在%d点开始, 时长%d学时' %(className, Time, duaration))
```

AI程序设计在6点开始，时长3学时

print() 函数

- str.format() 函数

```
print(' {} {}'.format(a,b))
```

className='AI程序设计'
Time=6
duaration=3

[26]: `print('{}在{}点开始，时长{}学时'.format(className, Time, duaration))`

AI程序设计在6点开始，时长3学时

[27]: `print('{}在{}点开始，时长{}学时，{}加油'.format(className, Time, duaration))`

AI程序设计在6点开始，时长3学时，AI程序设计加油

[37]: `print('{}在{}点开始，时长{}学时，\n{}加油'.format(classname=time, className=Time, classname=duaration))`

AI程序设计在6点开始，时长3学时，AI程序设计加油

[24]: `print('{}在{}点开始，时长{}学时'.format(className, Time, duaration))`

AI程序设计在6点开始，时长3学时

输入与输出

- Input
- Python提供了 `input()` 内置函数从标准输入读入一行文本，默认的标准输入是键盘
- <变量> = `input(<提示性文字>)`

In [*]: `className=input('请输入课程名称: ')`

请输入课程名称: AI程序 ×

In []:

In [39]: `className=input('请输入课程名称: ')`

请输入课程名称: AI程序设计

In [40]: `className`

Out[40]: 'AI程序设计'

- 计算机导论
- 编程导论
- 人工智能导论
- 人工智能与Python
- Python输入输出