

程序设计 Programming

Lecture 1: 程序设计与C语言简介



课程基本信息

- 办公室：数学馆东115
- Email: xslu@dase.ecnu.edu.cn
- 理论课：周一6、7，周五8、9（文史楼215）
- 实验课：周一11、12，周三11、12（数学馆西113）
- 答疑：周四14:00~16:00

助教、课程群

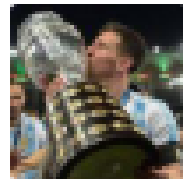
- 孙畅煊



Zoom

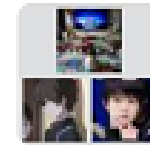
个人名片

- 陈郅睿



mesuarez

个人名片



2022程序设计课群



该二维码7天内(9月21日前)有效, 重新进入将更新

授课、学习方式

• 翻转课堂

- ✓ 水杉在线：课件、视频、作业、测验, <https://www.shuishan.net.cn>
- ✓ 中国大学MOOC: <https://www.icourse163.org/course/ZJU-1001614008>
- ✓ 线下课堂：重点内容讲解、答疑讨论、知识拓展



水杉在线



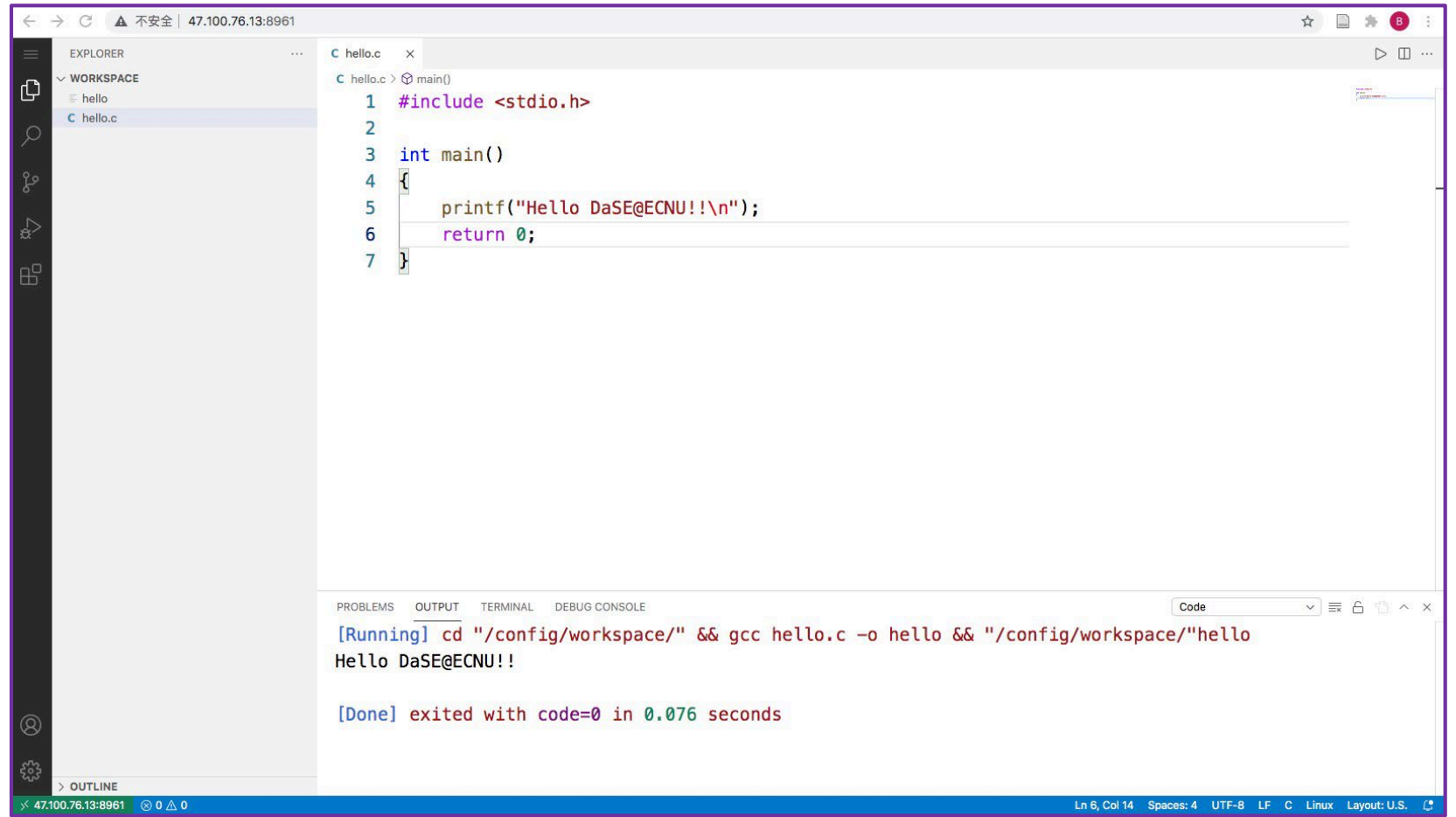
中国大学MOOC

水杉在线

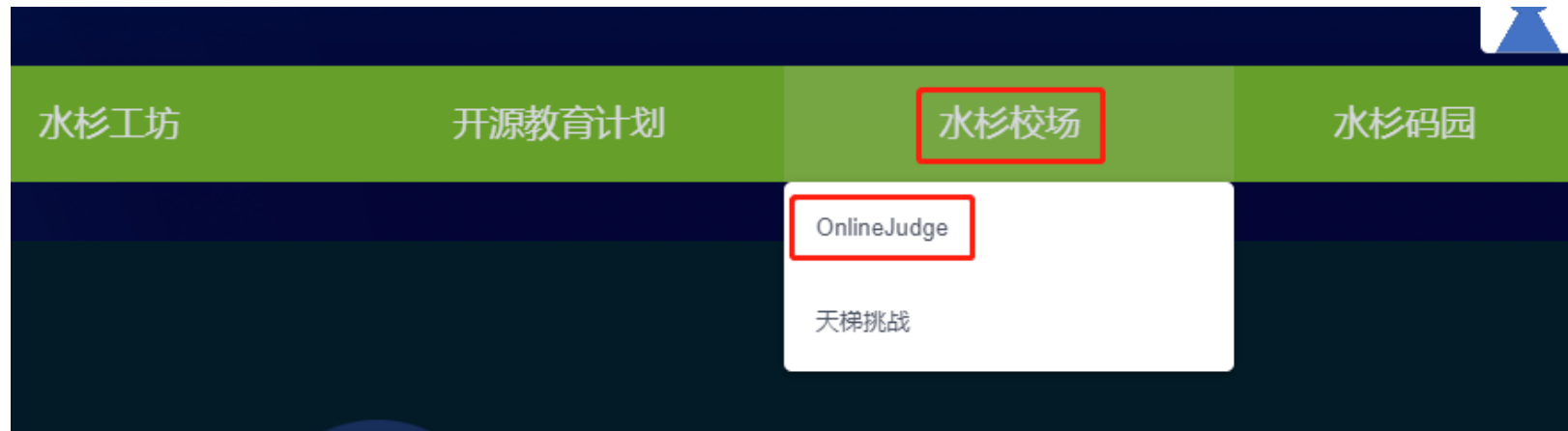
- 水杉学堂：下载课件、观看视频
- 水杉工坊：在线编程工具（VSCode）
- 水杉校场：完成作业与测验



水杉工坊: VSCode



水杉校场



Judge Field Home Problems **Contests** Status

All Contests

《C语言程序设计》第二章判断练习

2021-9-8 00:00 3 months ACM

《C语言程序设计》第一章计算练习

2021-9-6 00:00 3 months ACM

Problems List

#	Title	Total	AC Rate
2-1	求整数均值	545	20.55%
2-2	阶梯电价	333	31.83%
2-3	序列求和	502	20.92%
2-4	求交错序列前n项的和	317	32.81%
2-5	平方根求和	228	46.49%
2-6	求给定序列前n项的和 ($1!+2!+3!+\dots+n!$)	277	37.55%

学习方法

- 不要拘泥书本，开始写代码！
- 勤写代码是学习程序设计的最佳方式！
- 回头再看理论知识，温故而知新！

Just start and don't keep it only theoretical. Start building something. We learn best when we play with a system.

—Vishal Sikka



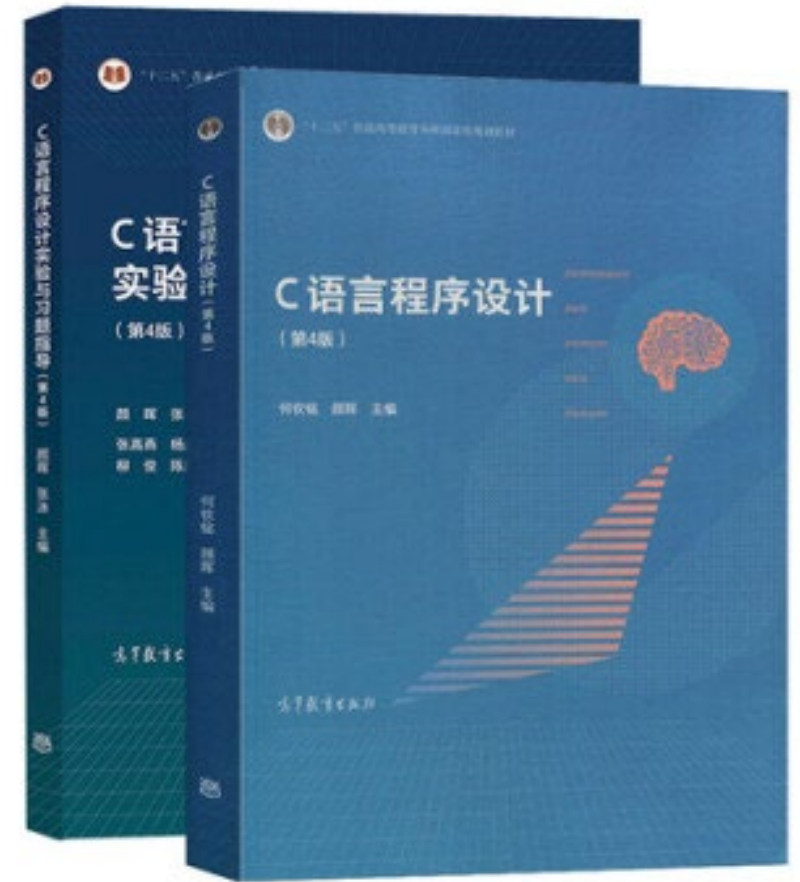
考核方式

- 平时成绩 (80%)
 - ✓ 考勤: 10%
 - ✓ 在线作业: 10%
 - ✓ 两次在线测验: 60%
(3小时8题)
- 期末笔试 (20%)
 - ✓ 理论知识和手写代码



参考书目和资料

- 何钦铭，颜晖。《C语言程序设计》第4版。高等教育出版社。
- K. N. King. 'C Programming: A Modern Approach'.
 - ✓ 吕秀锋 译。《C语言程序设计现代方法》。人民邮电出版社。
- 技术论坛：StackOverflow (SO), CSDN



Questions?

1. 程序设计基本概念

什么是程序设计 (Programming) ?

- 程序设计是指设计和编写一段计算机可以执行的程序，使得计算机能够完成某项特定任务。
 - ✓ 计算机任务：加减乘除、数据分析、数据库系统，图像/语音识别等等
 - ✓ 设计程序：问题、架构、算法、数据结构，等等
 - ✓ 编写程序 (coding)：使用程序语言 (programming language)

```

31
32 self.file = None
33 self.fingerprints = set()
34 self.logdups = True
35 self.debug = debug
36 self.logger = logging.getLogger(__name__)
37 if path:
38     self.file = open(os.path.join(path, "requests_
39     self.file.seek(0)
40     self.fingerprints.update(e.response)
41
42 @classmethod
43 def from_settings(cls, settings):
44     debug = settings.getbool("SUPERSITE_DEBUG")
45     return cls(job_dir(settings), debug)
46
47 def request_seen(self, request):
48     fp = self.request_fingerprint(request)
49     if fp in self.fingerprints:
50         return True
51     self.fingerprints.add(fp)
52     if self.file:
53         self.file.write(fp + os.linesep)
54
55 def request_fingerprint(self, request):
56     return request_fingerprint(request)

```

为什么要学程序设计？



鸡汤

- 改变你的思维方式
- 改变我们的世界

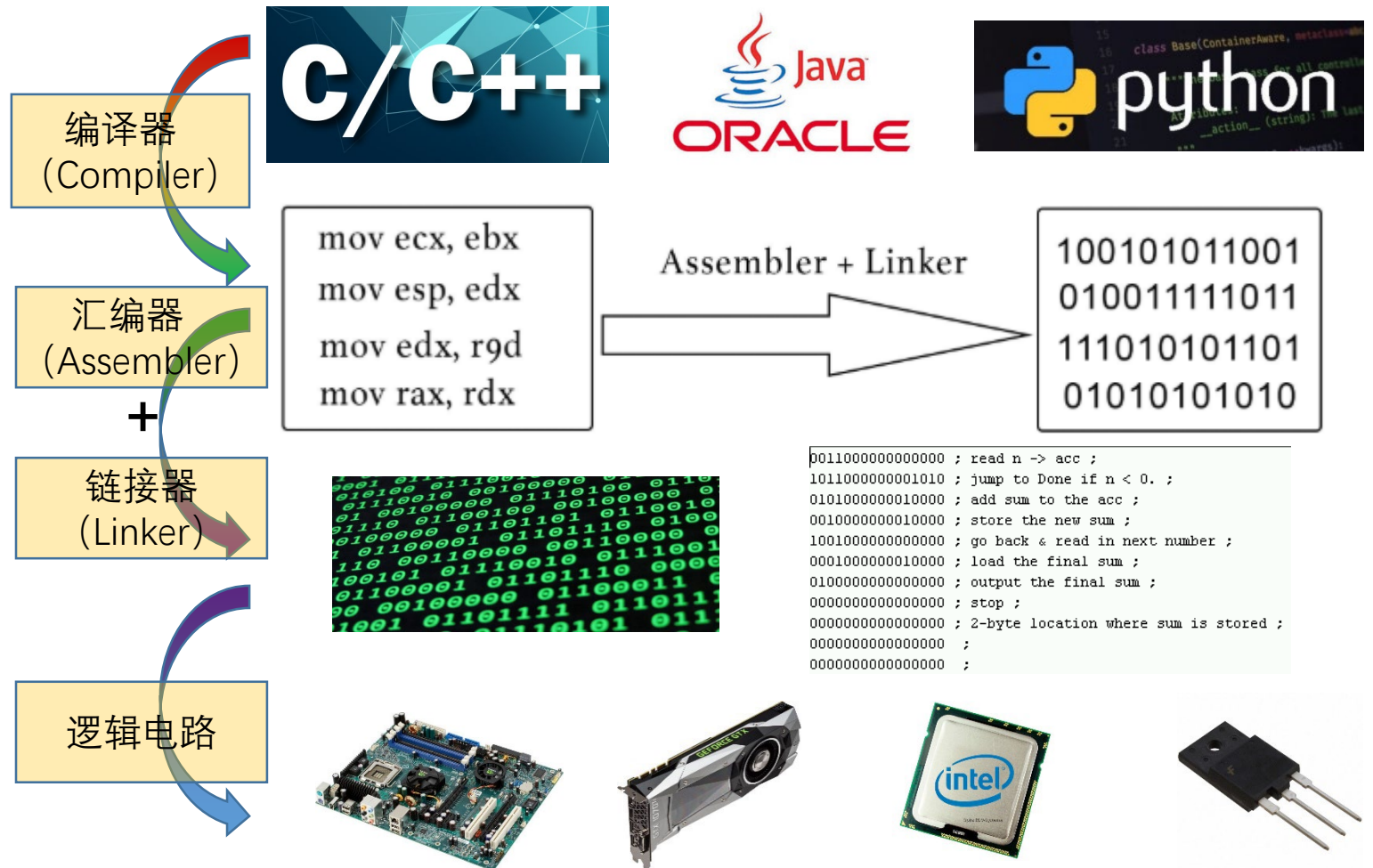


现实

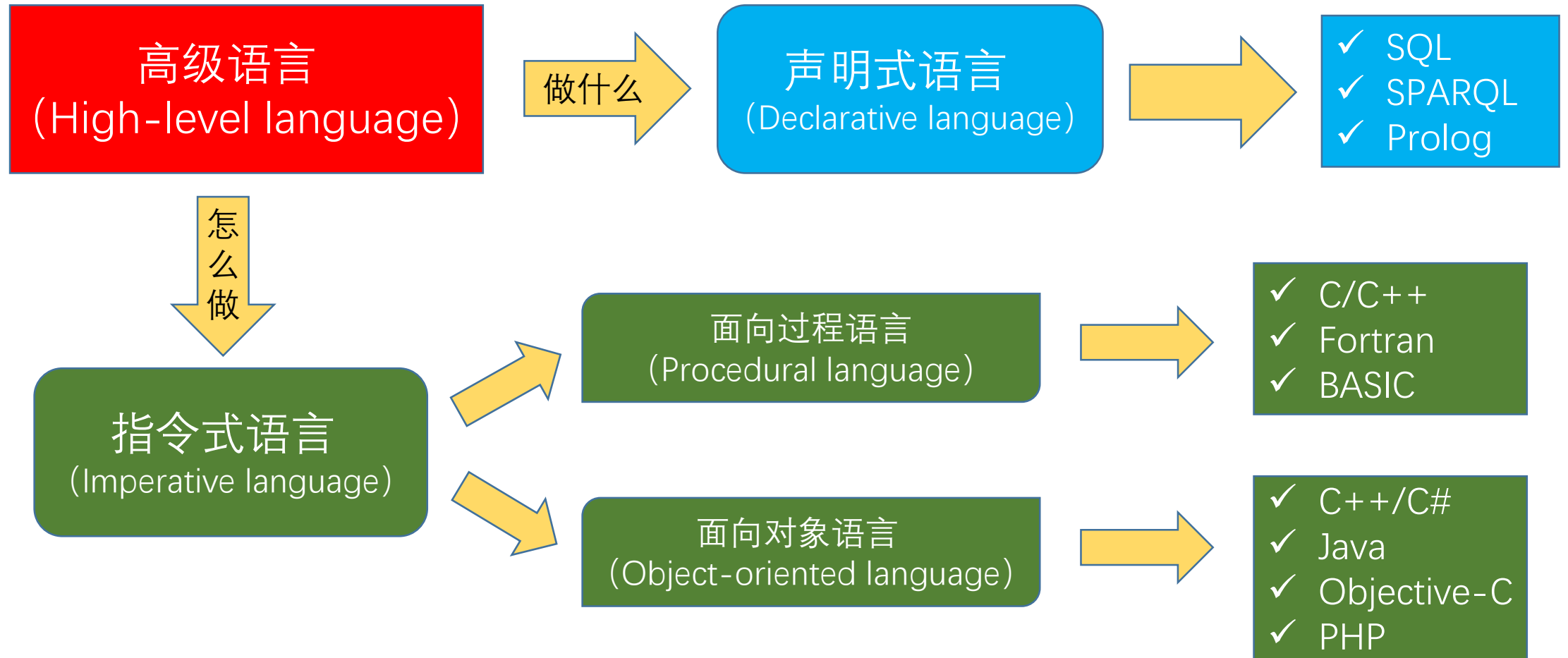
- 数据专业谋生的基本工具
- 世界将被改变

程序设计语言

抽象化



高级语言分类



C语言

- C is a general-purpose, procedural computer programming language supporting structured programming, lexical scope, and recursion, while a static type system prevents unintended operations.

C语言

- C is a **general-purpose, procedural** computer programming language supporting **structured programming**, **lexical scope**, and **recursion**, while a **static type system** prevents unintended operations.
- 通用的：底层硬件、系统产品、数据分析、人工智能
- 面向过程：先做什么，后做什么（面向对象：A做什么，B做什么）
- 结构化编程：控制结构、区块、子程序
- 词法作用域：所有变量都有自己的作用域，超出作用域则无效
- 递归：将大问题分解为相同类型的子问题
- 静态类型系统：在程序运行前检查类型问题

C语言： 输出斐波那契数列前n项

- 1, 1, 2, 3, 5, 8, 13, ...

```
1  #include <stdio.h>
2
3  long fib(int n)
4  {
5      if(n == 0 || n == 1)
6          return 1;
7      else
8          return (fib(n-1)+fib(n-2));
9  }
10
11 int main()
12 {
13     int i;
14
15     for(i = 0; i < 8; i++)
16         printf("%ld ", fib(i));
17     printf("\n");
18
19     return 0;
20 }
```

控制结构

C语言： 输出斐波那契数列前n项

• 1, 1, 2, 3, 5, 8, 13, ...

```
1  #include <stdio.h>
2
3  long fib(int n)
4  {
5      if(n == 0 || n == 1)
6          return 1;
7      else
8          return (fib(n-1)+fib(n-2));
9  }
10
11 int main()
12 {
13     int i;
14
15     for(i = 0; i < 8; i++)
16         printf("%ld ", fib(i));
17     printf("\n");
18
19     return 0;
20 }
```

控制结构

变量n的作用域

C语言：输出斐波那契数列前n项

- 1, 1, 2, 3, 5, 8, 13, ...

```

1  #include <stdio.h>
2
3  long fib(int n)
4  {
5      if(n == 0 || n == 1)
6          return 1;
7      else
8          return (fib(n-1)+fib(n-2));
9  }
10
11 int main()
12 {
13     int i;
14
15     for(i = 0; i < 8; i++)
16         printf("%ld ", fib(i));
17     printf("\n");
18
19     return 0;
20 }

```

控制结构

变量n的作用域

递归函数

C语言：输出斐波那契数列前n项

- 1, 1, 2, 3, 5, 8, 13, ...

```

1  #include <stdio.h>
2
3  long fib(int n)
4  {
5      if(n == 0 || n == 1)
6          return 1;
7      else
8          return (fib(n-1)+fib(n-2));
9  }
10
11 int main()
12 {
13     int i;
14
15     for(i = 0; i < 8; i++)
16         printf("%ld ", fib(i));
17     printf("\n");
18
19     return 0;
20 }

```

控制结构

变量n的作用域

递归函数

类型检查

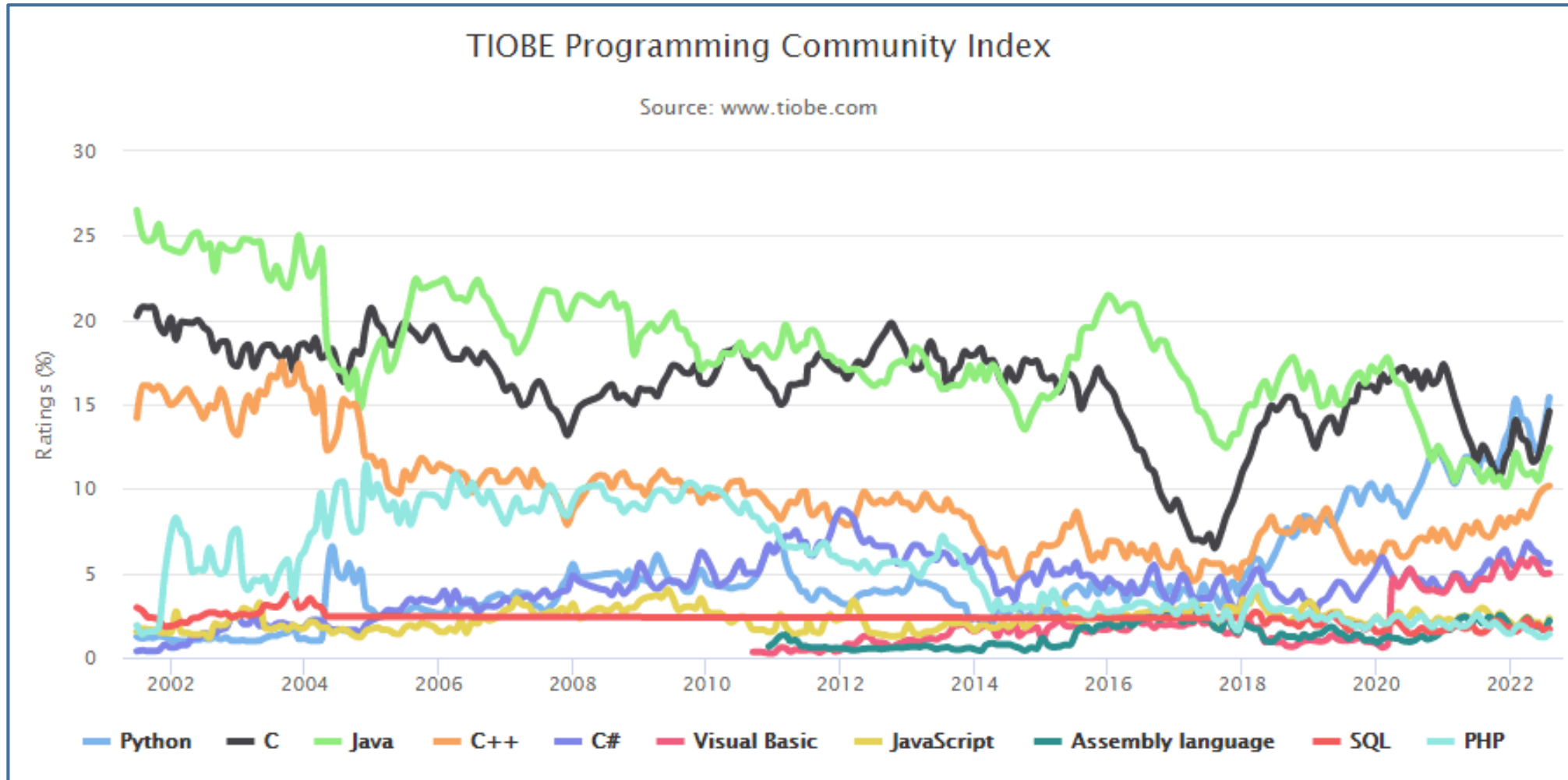
C语言的发展历史

时间	C标准	事件
1972年	Birth	贝尔实验室Dennis Ritchie和Ken Thompson等人开发Unix系统时发明了C语言
1978年	K&R C	Brian Kernighan和Dennis Ritchie出版了第一本C语言教程《The C Programming Language》
1989年	ANSI C	美国国家标准学会和国际标准化组织（ISO）发布了C语言的标准（标准C）
1999年	C99	加入了内联函数、新的数据类型、可变长数组等新特性
2011年	C11	加入了通用宏、匿名结构、多线程和边界检查函数等新特性
2018年	C18	最新的C语言标准，对C11的缺陷进行了修订

C语言的发展历史

时间	C标准	事件
1972年	Birth	贝尔实验室Dennis Ritchie和Ken Thompson等人开发Unix系统时发明了C语言
1978年	K&R C	Brian Kernighan和Dennis Ritchie出版了第一本C语言教程《The C Programming Language》
1989年	ANSI C	美国国家标准学会和国际标准化组织（ISO）发布了C语言的标准（标准C）
1999年	C99	加入了内联函数、新的数据类型、可变长数组等新特性
2011年	C11	加入了通用宏、匿名结构、多线程和边界检查函数等新特性
2018年	C18	最新的C语言标准，对C11的缺陷进行了修订

TIOBE programming community index



为什么要学C语言？

- 简而言之，C语言通用、易懂（相对于底层语言），并且涉及很多计算机底层和基础知识（相对于更高级语言）
 - ✓ 语法简单，容易上手
 - ✓ 非常少量的库函数
 - ✓ 高效运行
 - ✓ 帮助了解计算机基础理论知识
 - ✓ 可用于嵌入式编程

为什么要学C语言？

- 简而言之，C语言通用、易懂（相对于底层语言），并且涉及很多计算机底层和基础知识（相对于更高级语言）
 - ✓ 语法简单，容易上手
 - ✓ 非常少量的库函数

学好C语言，再学习其他语言就变得很简单！

- ✓ 帮助了解计算机基础理论知识
- ✓ 可用于嵌入式编程

C语言 (C++) 能做什么

- 编写操作系统
 - ✓ Windows\Linux\Mac\iOS\Android
- 实现其他语言底层架构
 - ✓ JVM, Python编译器
- 实现数据库系统
 - ✓ Oracle\Mysql\PostgreSQL
- 嵌入式系统
- 桌面应用
- 人工智能
- ○ ○ ○

After All These Years, the World is Still Powered by C Programming



DANIEL ANGEL MUÑOZ TREJO

Daniel has created high-performance applications in C++ for large companies such as Dreamworks. He also excels with C and ASM (x86).



C语言的缺点

- 可能会漏洞百出
 - ✓C语言非常灵活，语法上限制相对较少，比如字符+整型
- 可能会难以理解
 - ✓很多语法特性可以结合使用，旁观者难以理解
- 可能会难以修改
 - ✓缺少标准化模块管理机制，大规模程序难以管理

International Obfuscated C Code Contest

- 国际模糊C代码大赛

International Obfuscated C code Contest

ntest

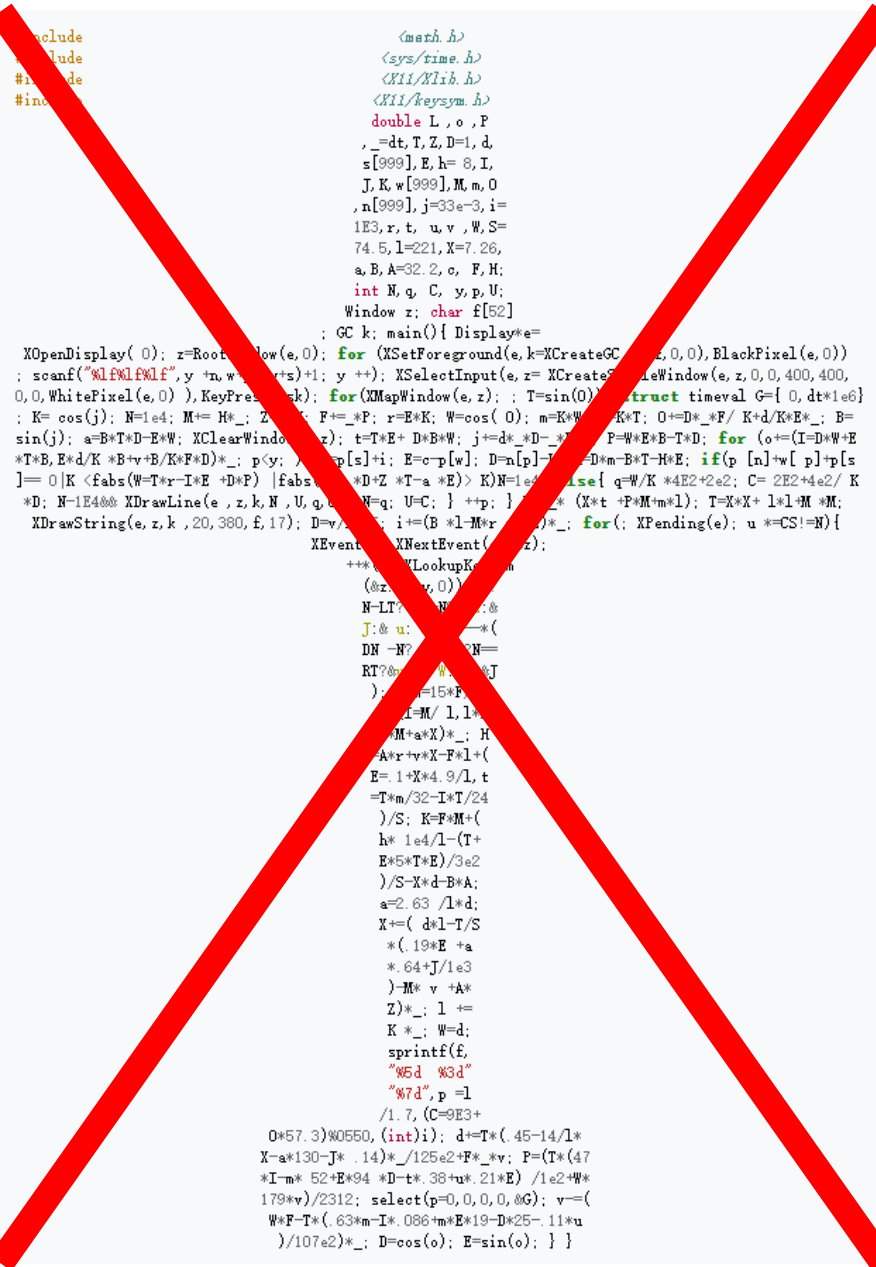
- 国际模糊C代码大赛

```
#include <math.h>
#include <sys/time.h>
#include <X11/Xlib.h>
#include <X11/keysym.h>
double L,o,P;
_,dt,T,Z,D=1,d;
s[999],E,h=8,I;
J,K,w[999],M,m,0;
,n[999],j=33e-3,i=
1E3,r,t,u,v,W,S=
74.5,l=221,X=7.26,
a,B,A=32.2,c,F,H;
int N,q,C,y,p,U;
Window z;char f[52];
:GC k;main(){Display*=
XOpenDisplay(0);z=RootWindow(e,0);for(XSetForeground(e,k=XCreateGC(e,z,0,0),BlackPixel(e,0))
:scanf("%lf%lf%lf",y,"n,w+y,y+s)+1;y++);XSelectInput(e,z=XCreateSimpleWindow(e,z,0,0,400,400,
0,0,WhitePixel(e,0)),KeyPressMask);for(XMapWindow(e,z);T=sin(0)){struct timeval G={0,dt*1e6}
;K=cos(j);N=1e4;M+=H*_;Z=D*K;F+=_P;r=E*K;W=cos(0);m=K*W;H=K*T;O+=D*_F/K+d/K*E*_;B=
sin(j);a=B*T*D-E*W;XClearWindow(e,z);t=T*E+D*B*W;j+=d*_D*_F*E;P=W*E*B-T*D;for(o+=(I=D*W+E
*T*B,E*d/K*B+v+B/K*F*D)*_;p<y:){T=p[s]+i;E=c-p[w];D=n[p]-L;K=D*m-B*T-H*E;if(p[n]+w[p]+p[s
]=0|K<fabs(W=T*r-I*E+D*F)|fabs(D=t*D+Z*T-a*E)>K)N=1e4;else{q=W/K*4E2+2e2;C=2E2+4e2/K
*D;N=1E4@@XDrawLine(e,z,k,N,U,q,C);N=q;U=C;}++p;}L+=*(X*t+P*M+m*1);T=X*X+1*1*M*M;
XDrawString(e,z,k,20,380,f,17);D=v/l*15;i+=(B*1-M*r-X*Z)*_;for(XPending(e);u*=CS!=N){
XEvent z;XNextEvent(e,&z);
++*((N=XLookupKeysym
(&z,key,0))-IT?
N-LT?UF-N?@E:@
J:@u:@h);--*(
DN-N?N-DT?N=
RT?@u:@W:@h:@J
);}m=15*F/L;
o+=(I-M/l,l*M
+I*M+a*X)*_;H
=A*r+v*X-F*1+(
E=1+X*4.9/l,t
=T*m/32-I*T/24
)/S;K=F*M+(
h*1e4/l-(T+
E*5*T*E)/3e2
)/S-X*d-B*A;
a=2.63/l*d;
X+=(d*l-T/S
*(.19*E+a
*.64+J/1e3
)-M*v+A*
Z)*_;l+=
K*_;W=d;
sprintf(f,
"%5d %3d"
"%7d",p,l
/1.7,(C=9E3+
0*57.3)%0550,(int)i);d+=T*(.45-14/l*
X-a*130-J*.14)*_/125e2+F*_v;P=(T*(47
*I-m*52+E*94*D-t*.38+u*.21*E)/1e2+W*
179*v)/2312;select(p=0,0,0,0,&G);v=(
W*F-T*(.63*m-I*.086+m*E*19-D*25-.11*u
)/107e2)*_;D=cos(o);E=sin(o);} }
```

International Obfuscated C code Contest

ntest

- 国际模糊C代码大赛



如何高效编写C语言程序

- 规避C语言的缺陷
- 使用工具
 - ✓ IDE (VSCode, 我们提供在线服务!)
 - ✓ 调试工具 (Debug, 未来某次课)
- 利用现有代码库
- 养成好的编程规范和风格
- 简洁, 不要“秀”编程技巧

2. 一个完整的C语言

预处理指令

```
1  /*****
2
3      This is my first C programme.
4
5  *****/
6
7  #include <stdio.h>
8
9  int fib(int n)
10 {
11     if(n == 0 || n == 1)
12         return 1;
13     else
14         return (fib(n-1)+fib(n-2));
15 }
16
17 int main()
18 {
19     int i;
20
21     for(i = 0; i < 8; i++)
22         printf("%d ", fib(i));
23     printf("\n");
24
25     return 0;
26 }
27
```

- 将stdio.h“头文件”中的信息包含到程序中
 - ✓以#开头
 - ✓C标准输入/输出库
 - ✓这里会用到printf函数
- C语言有大量类似stdio.h的头文件，包含了各种C标准库
 - ✓<stdlib.h>，基础工具
 - ✓<string.h>，字符串处理
 - ✓<math.h>，常用数学函数
- 其他预处理指令
 - ✓#define, #ifdef, #ifndef等等

函数

```
1  /*****
2
3      This is my first C programme.
4
5  *****/
6
7  #include <stdio.h>
8
9  int fib(int n)
10 {
11     if(n == 0 || n == 1)
12         return 1;
13     else
14         return (fib(n-1)+fib(n-2));
15 }
16
17 int main()
18 {
19     int i;
20
21     for(i = 0; i < 8; i++)
22         printf("%d ", fib(i));
23     printf("\n");
24
25     return 0;
26 }
27
```

- 完成特定任务的独立模块，接收参数，返回结果
 - ✓ fib: 程序员编写的函数
 - ✓ printf: 库函数
 - ✓ main: 主函数，C程序入口
- 函数定义
 - ✓ 函数头：返回值类型，函数名，参数和类型
 - ✓ 函数体：规定函数的功能
- 函数调用
 - ✓ 传递参数
 - ✓ 执行函数规定的功能

语句

```

1  /*****
2
3      This is my first C programme.
4
5  *****/
6
7  #include <stdio.h>
8
9  int fib(int n)
10 {
11     if(n == 0 || n == 1)
12         return 1;
13     else
14         return (fib(n-1)+fib(n-2));
15 }
16
17 int main()
18 {
19     int i;
20
21     for(i = 0; i < 8; i++)
22         printf("%d ", fib(i));
23     printf("\n");
24
25     return 0;
26 }
27

```

条件语句

循环语句

函数调用语句

返回语句

一个最简单的C程序语言

- 预处理指令
 - ✓ #include, #define, #ifdef, #ifndef等等
- 函数
 - ✓ 主函数, 库函数, 程序员自定义函数
- 语句
 - ✓ 条件语句, 循环语句, 返回语句等等

预处理指令

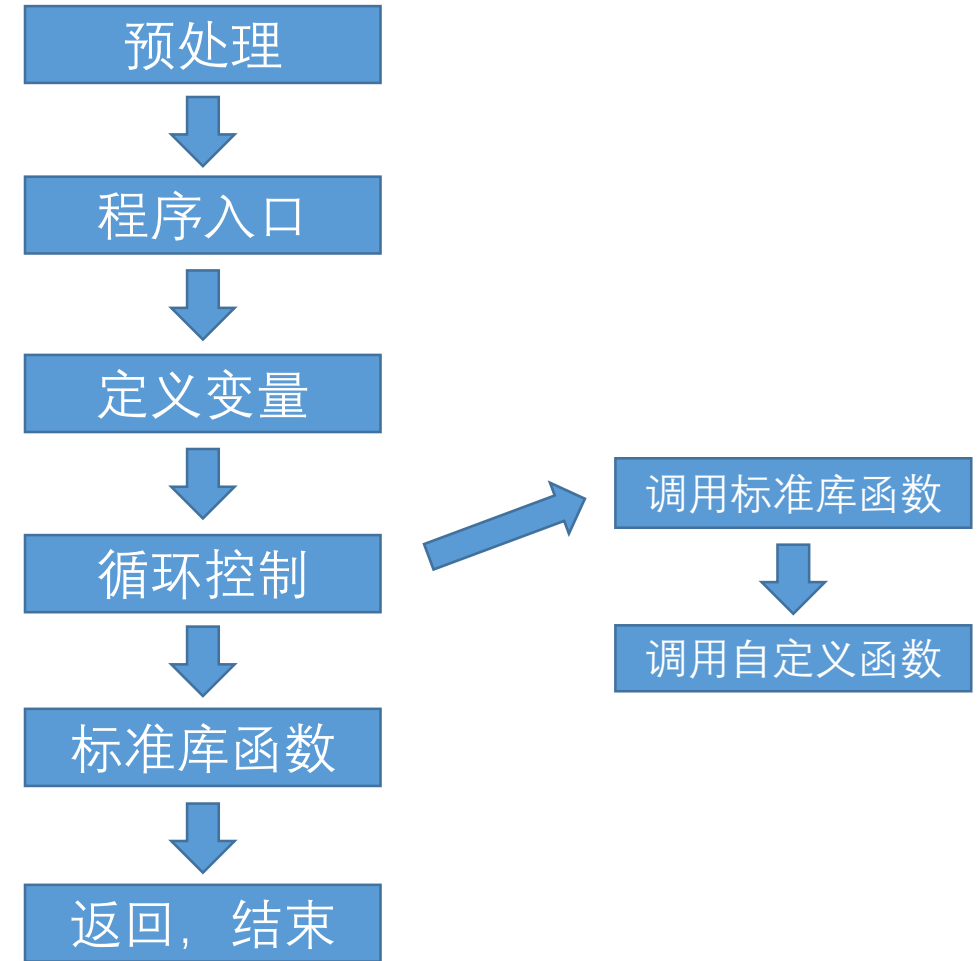
自定义函数（可以没有）

```
main ()
{
    语句
}
```

C语言运行过程：输出斐波那契数列前n项

• 1, 1, 2, 3, 5, 8, 13, ...

```
1 #include <stdio.h>
2
3 long fib(int n)
4 {
5     if(n == 0 || n == 1)
6         return 1;
7     else
8         return (fib(n-1)+fib(n-2));
9 }
10
11 int main()
12 {
13     int i;
14
15     for(i = 0; i < 8; i++)
16         printf("%ld ", fib(i));
17     printf("\n");
18
19     return 0;
20 }
```



注释

```

1  /******
2
3      This is my first C programme.
4
5  *****/
6
7  #include <stdio.h>
8
9  int fib(int n)
10 {
11     if(n == 0 || n == 1)
12         return 1;
13     else
14         return (fib(n-1)+fib(n-2));
15 }
16
17 int main()
18 {
19     int i;
20
21     for(i = 0; i < 8; i++)
22         printf("%d ", fib(i));
23     printf("\n");
24
25     return 0;
26 }
27

```

- 不会被执行的代码说明
 - ✓ 程序名, 日期, 作者, 用途
 - ✓ 模块功能
 - ✓ 函数或语句的用途
 - ✓ 其他必要的说明
- C风格: /*开头, */结束
- C++风格: //开头, 换行结束

注释

```
1  /*****
2
3      计算斐波那契数列前n项
4
5  *****/
6
7  #include <stdio.h>
8
9  /*****递归计算斐波那契数列第n项*****/
10 int fib(int n)
11 {
12     if(n == 0 || n == 1)
13         return 1;
14     else
15         return (fib(n-1)+fib(n-2));
16 }
17
18 int main()
19 {
20     int i;
21
22     for(i = 0; i < 8; i++) //循环打印斐波那契数列前8项
23         printf("%d ", fib(i));
24     printf("\n");
25
26     return 0;
27 }
28
```

程序功能

函数功能

语句功能， C++风格

注释

```
1  /*****  
2  
3      计算斐波那契数列前n项  
4  
5  *****/  
6  
7  #include <stdio.h>  
8  
9  /*****递归计算斐波那契数列第n项*****/  
10 int fib(int n)  
11 {  
12     if(n == 0 || n == 1)  
13         return 1;  
14     else  
15         return (fib(n-1)+fib(n-2));
```

程序功能

函数功能

养成勤写注释的习惯，对编写大型程序至关重要！！

```
23     printf("%d ", fib(i));  
24     printf("\n");  
25  
26     return 0;  
27 }  
28
```

语句功能，C++风格

Lecture 2

- C语言基本概念与元素