程序设计
Programming

Lecture 6: Linux和GCC简介

DaSE
Data Science
& Engineering

EAST CHINA NORMAL UNIVERSITY
華東師範大學

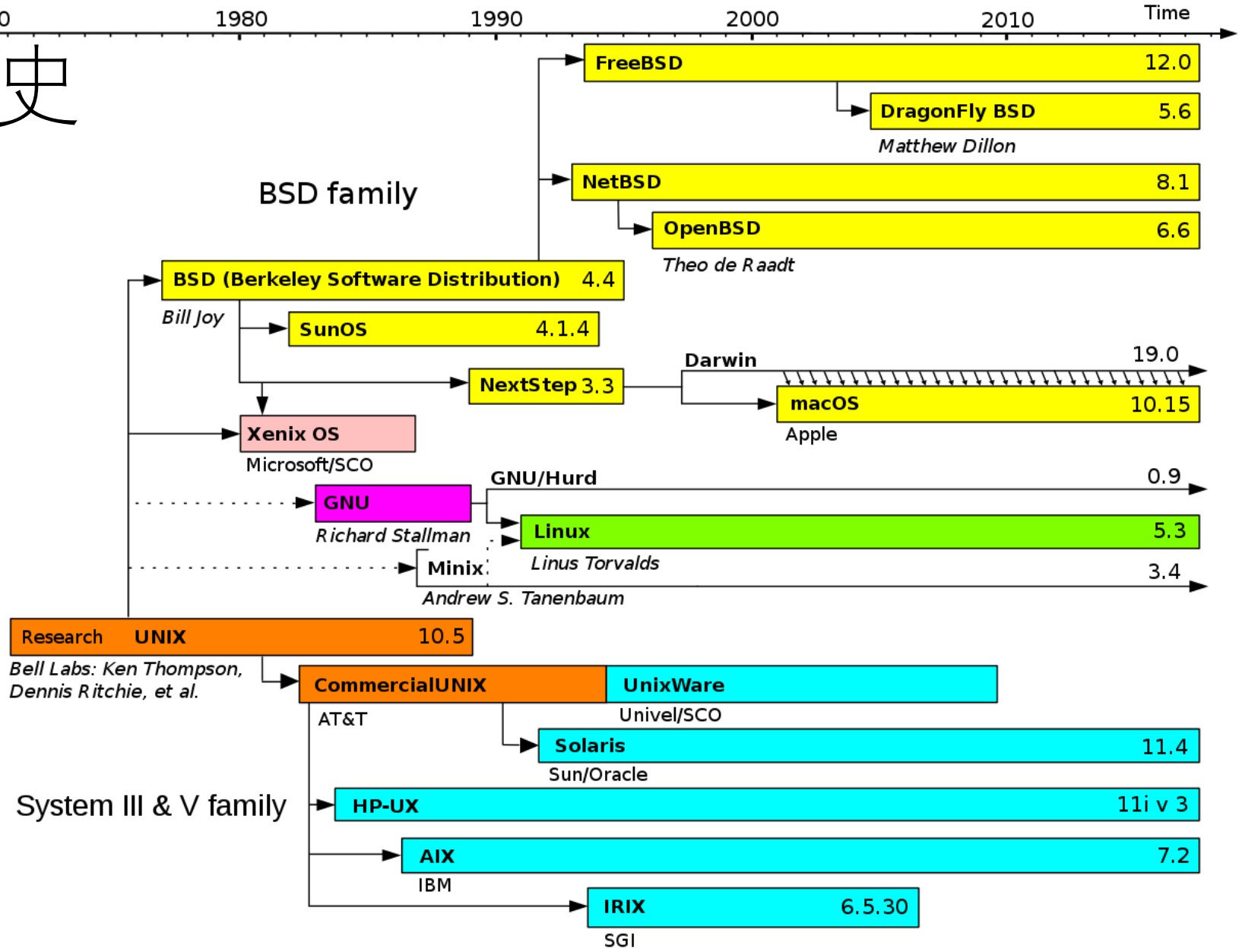
Linux操作系统

- Linux是一系列基于Linux内核、类似于Unix操作系统的开源操作系统
  - ✓第一个Linux内核于1991年发布
  - ✓一个Linux操作系统包括Linux内核，以及系统软件和库文件，后者大部分由GNU开源项目发开
  - ✓常见的Linux发行版本： Red Hat/Centos, Ubuntu, Debian, Fedora, Suse

- 与之对应：Mac OS 1984年，Windows 1985年

操作系统简史

1973


1970
1980
1990
2000
2010
Time
BSD family
FreeBSD 12.0
DragonFly BSD 5.6
Matthew Dillon
NetBSD 8.1
OpenBSD 6.6
Theo de Raadt
BSD (Berkeley Software Distribution) 4.4
Bill Joy
SunOS 4.1.4
NextStep 3.3
Darwin 19.0
macOS 10.15
Apple
Xenix OS
Microsoft/SCO
GNU/Hurd 0.9
GNU
Richard Stallman
Linux 5.3
Linus Torvalds
Minix 3.4
Andrew S. Tanenbaum
Research UNIX 10.5
Bell Labs: Ken Thompson, Dennis Ritchie, et al.
CommercialUNIX
UnixWare
Univel/SCO
AT&T
Solaris 11.4
Sun/Oracle
System III & V family
HP-UX 11i v 3
AIX 7.2
IBM
IRIX 6.5.30
SGI

操作系统简史

1984

1973

Unix-like

操作系统简史

1973

unix

Unix-like

1984

Mac™ OS

MS-DOS

1985

MICROSOFT® WINDOWS

操作系统简史

操作系统简史

1973

unix

1984

Mac OS

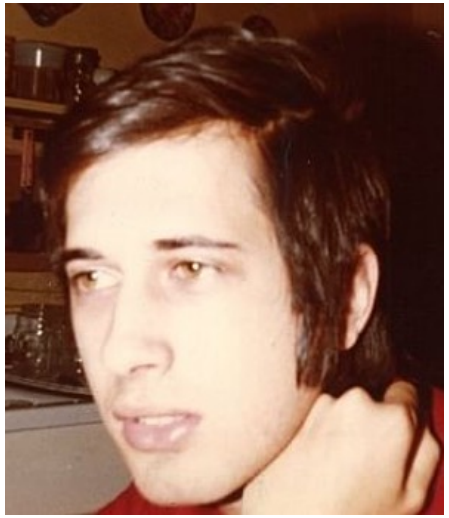
1985

MICROSOFT WINDOWS

Unix-like

1983

GNU



操作系统简史

1973

unix

1984

Mac OS

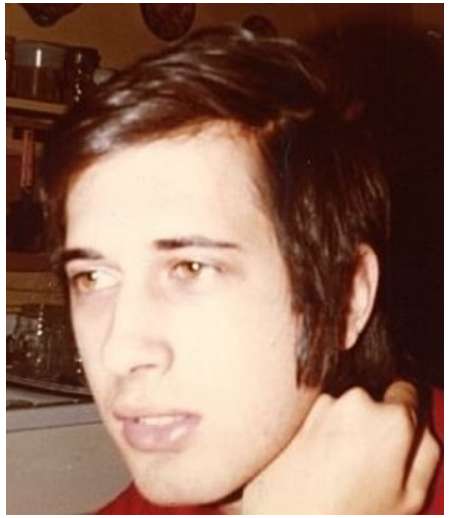
1985

MICROSOFT WINDOWS

Unix-like

1983

GNU



Unix-like

1991

Linux

操作系统简史

1973

unix

1984

Mac OS

1985

MICROSOFT WINDOWS

Unix-like

1983

GNU

Unix-like

1991

Linux

GNU/Linux

GNU Toolchain

- 开发操作系统和应用程序的一套工具，包括
  - ✓GNU Compiler Collection (GCC)：支持多种语言的编译器套件
  - ✓GNU Make：自动化编译和应用程序构建工具
  - ✓GNU Binutils：可执行工具套件，如汇编器（as）和链接器（ld）
  - ✓GNU Debugger (GDB)：调式工具
  - ✓GNU Autotools
  - ✓GNU Bison

回想：L01 程序语言编译和构建过程

高级语言
(High-level language)

汇编语言
(Assembly language)

机器语言（二进制文件）
(Machine language)

编译器
(Compiler)

汇编器
(Assembler)

+

链接器
(Linker)


```
mov ecx, ebx
mov esp, edx
mov edx, r9d
mov rax, rdx
```

Assembler + Linker

```
100101011001
010011111011
111010101101
01010101010
```

```
0011000000000000 ; read n -> acc ;
1011000000001010 ; jump to Done if n < 0. ;
0101000000010000 ; add sum to the acc ;
0010000000010000 ; store the new sum ;
1001000000000000 ; go back & read in next number ;
0001000000010000 ; load the final sum ;
0100000000000000 ; output the final sum ;
0000000000000000 ; stop ;
0000000000000000 ; 2-byte location where sum is stored ;
0000000000000000  ;
0000000000000000  ;
```

GCC：最早意为 GNU C Compiler（了解）

- 是GNU开发的C语言编译器，gcc
  - ✓C++程序的编译器：g++

- 基本命令形式
  - ✓gcc c代码文件

- 例如：gcc –o test.out test.c

GNU Make:自动化编译和程序构建(简单了解)

- 在MakeFile中编写大型项目的编译规则
  - ✓编译目标：前提1 前提2 前提3 …
  - ✓ gcc/g++ Command
- 使用make命令来执行自动编译和程序构建
  - ✓Make 编译目标（可省略）
- gcc和make参考Tutorial
  - ✓https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc_make.html

GDB： GNU Debugger（简单了解）

- 先以debugger模式编译出二进制文件
  - ✓gcc –g c代码文件
- 然后进入debugger环境
  - ✓gdb 二进制文件
- 参考Tutorial
  - ✓https://www.geeksforgeeks.org/gdb-command-in-linux-with-examples/

VSCode中调试（debug，掌握）

- 一、设置断点
- 二、点击左侧“run and debug”（“运行和调试”）
- 三、使用调试按钮，观察变量变化

VSCode中调试：设置断点

- 直接点击要观察的代码行的最左侧，程序运行到此行会暂停

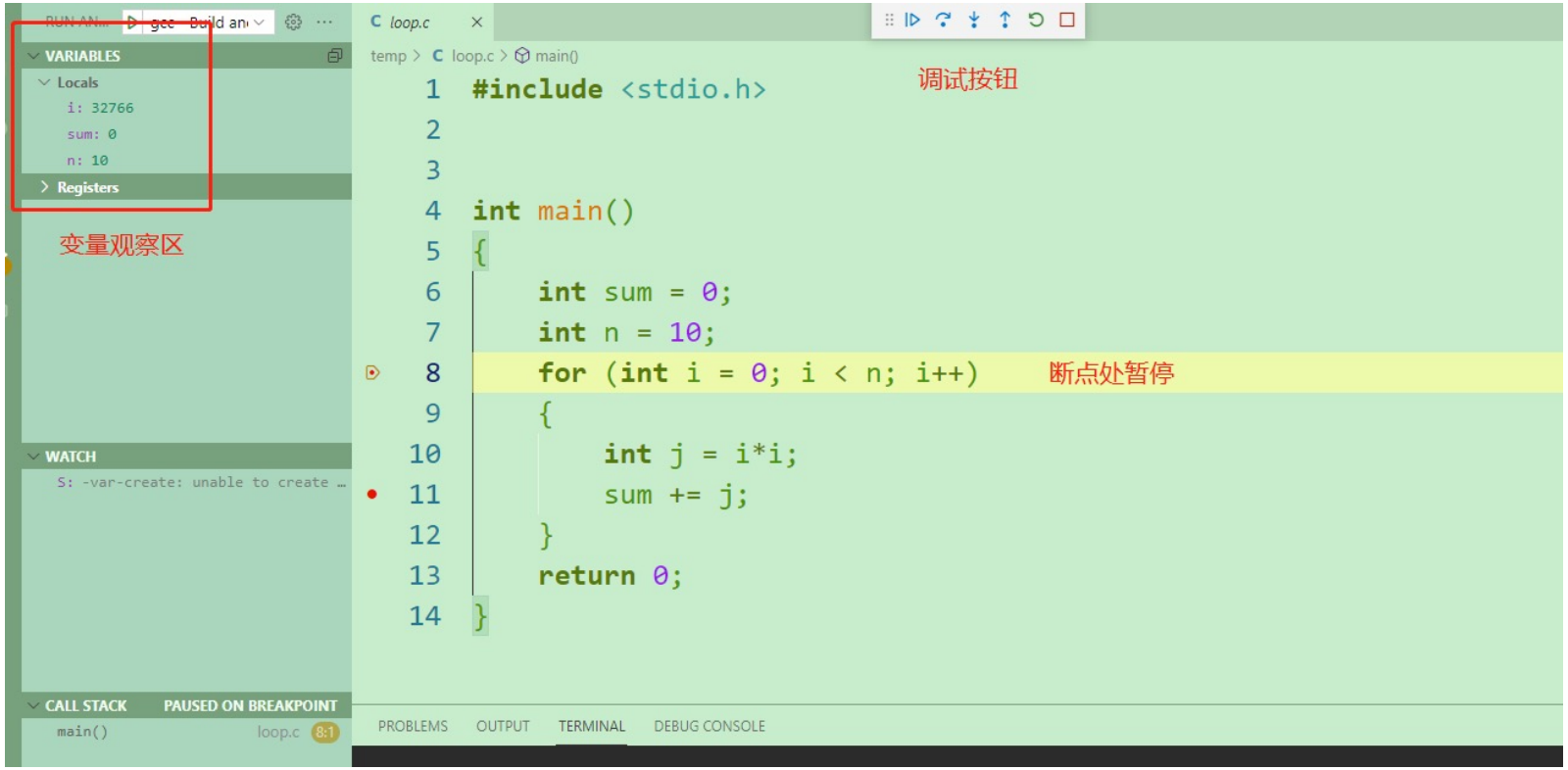
```
4  int main()
5  {
6      int sum = 0;
7      int n = 10;
8      for (int i = 0; i < n; i++)
9      {
10         int j = i*i;
11         sum += j;
12     }
13     return 0;
14 }
```

断点

断点

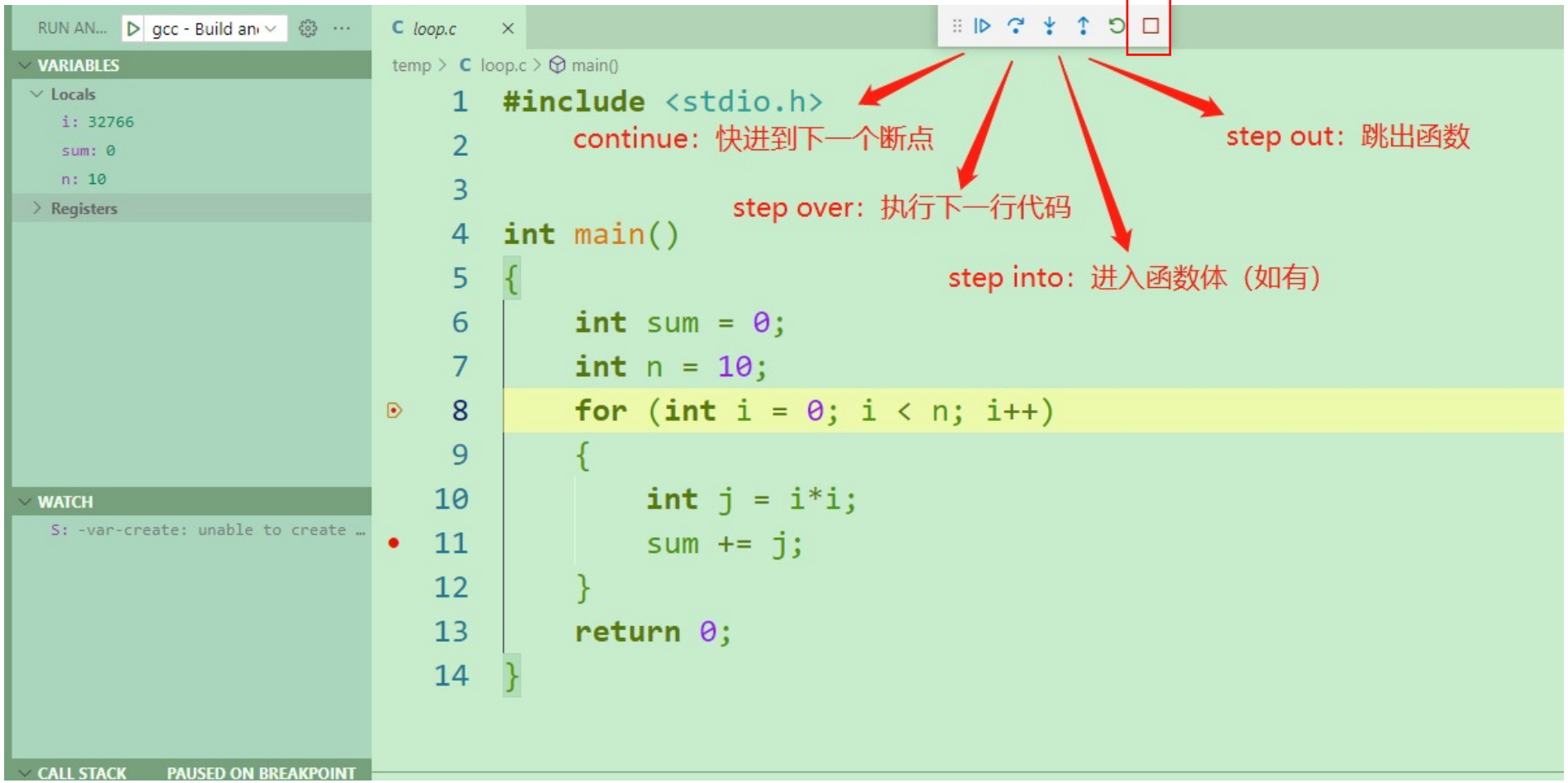
VSCode中调试：运行和调试

- 点击左侧“run and debug”，也可以直接摁快捷键F5



VSCode中调试：调式按钮

调试完毕，退出debug



Linux常用命令

- 修改密码：passwd
- 导航：pwd，cd
- 浏览：ls，less，file，cat，grep
- 文件操作：cp，mv，mkdir，rm
- 帮助：type，which，help，man
- I/O重定向：>，>>
- 管道：|
- 推荐自学网站：http://linuxcommand.org/lc3_learning_the_shell.php

Learning the Shell

Previous | Contents | Next

Why Bother?

Why do you need to learn the command line anyway? Well, let me tell you a story. A few years ago we had a problem where I used to work. There was a shared drive on one of our file servers that kept getting full. I won't mention that this legacy operating system did not support user quotas; that's another story. But the server kept getting full and it stopped people from working. One of our software engineers spent the better part of a day writing a C++ program that would look through all the user's directories and add up the space they were using and make a listing of the results. Since I was forced to use the legacy OS while I was on the job, I installed a Linux-like command line environment for it. When I heard about the problem, I realized I could do all the work this engineer had done with this single line:

```
du -s * | sort -nr > $HOME/user_space_report.txt
```

Graphical user interfaces (GUIs) are helpful for many tasks, but they are not good for all tasks. I have long felt that most computers today are not powered by electricity. They instead seem to be powered by the "pumping" motion of the mouse! Computers were supposed to free us from manual labor, but how many times have you performed some task you felt sure the computer should be able to do but you ended up doing the work yourself by tediously working the mouse? Pointing and clicking, pointing and clicking.

I once heard an author say that when you are a child you use a computer by looking at the pictures. When you grow up, you learn to read and write. Welcome to Computer Literacy 101. Now let's get to work.

Contents

Previous | Contents | Top | Next