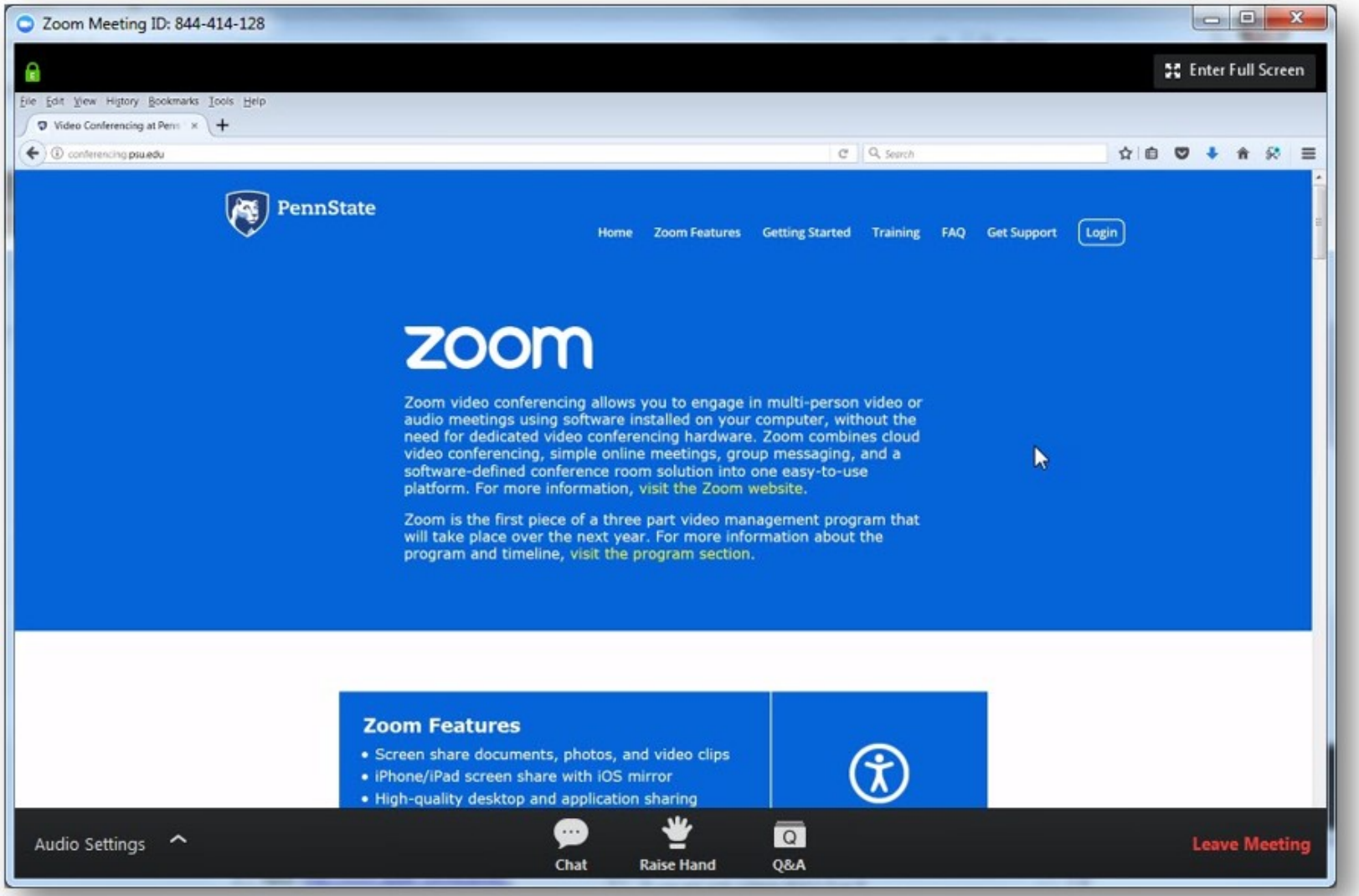
University of Essex

C++ Programming

01- Introduction

Zoom webinar guidelines



Module Schedule

- Two lectures (Monday 13.00-13.50 and Tuesday 16.00-16.50) each week in the autumn term
- We have pre-recorded lectures focused on the theory
- The live lectures will hold tutorial-style lectures plus Q&As
- One 2-hour lab at 13.00 on Thursdays.

(Note that the labs start in week 3 – there are none in week 2)

- There will also be two revision lectures in the summer term.

Assessment

- One two-hour examination in May/June (60% of the module credit)
- Two programming assignments to be submitted by 23/11/2020 and 20/01/2021 (20% each)
- **(Note that this module has no week 6 test.)**

Recommended Reading

- The main recommended text for this module is
  - *Thinking in C++*, B. Eckel, Volume 1, 2nd edition (Prentice Hall, 2000)
- available for free download at
  - https://www.micc.unifi.it/bertini/download/programmazione/TICPP-2nd-ed-Vol-one-printed.pdf

  Alternative books include
  - *C++: How to Program,* P.J. Deitel and H.M. Deitel, 10th edition (Pearson, 2016)
  - *C++ for Java Programmers*, T. Budd (Addison Wesley, 1999), and the definitive reference to the original version of C++
  - *The C++ Programming Language*, B. Stroustrup, 3rd edition (Addison Wesley, 2000) .

How to succeed in this module?

- Watch all pre-recorded videos before the lectures
- Go through the examples given in the tutorials
- Complete the labs within a few days (mostly)
- Practice, practice, practice!!!
- Ask questions
- Communicate (with me and colleagues)!!
- Work on your assignments weekly (don’t leave to do it right before the deadline)
- Make sure you attend the review lectures!!
- Make sure to revisit all available material (videos, slides, etc) before the exam

Why learn C++

- C++ is more powerful and efficient than other high-level programming languages (although more complicated).
- C and C++ are the most widely used programming languages for robotics and games, and for writing compilers and operating systems and drivers of hardware devices.
- C++ is built based on C, and C syntax is used in C++
- Consequently a knowledge of C++ is a big advantage in the jobs market.

Application written in C++

- OS
  - Macos, Linux, Windows
- Business and web applications
  - Microsoft Office
- Large-scale graphics applications
  - Adobe photoshop
- Web browsers and web applications
  - Firefox, google chrome
- Other programming languages
  - Java, Python, PhP, Pearl, etc

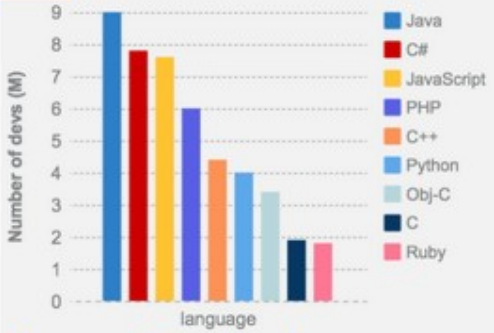
All about C++

#1

~4.4 million C++ devs
~1.9 million C devs

There are 4.4 million C++ developers and nearly 2 million C developers in the world.

#2



As many C++ developers as Python developers

We've analyzed a range of sources to estimate the number of worldwide developers using the most popular languages.

C++ is on par with Python, while the adoption of C is similar to that of Ruby.

#3

C++ developers by world region

EMEA and Asia-Pacific are the most densely populated regions with regard to developers in general and C++ developers in particular.

#4

Where C++ is relatively ahead of other languages

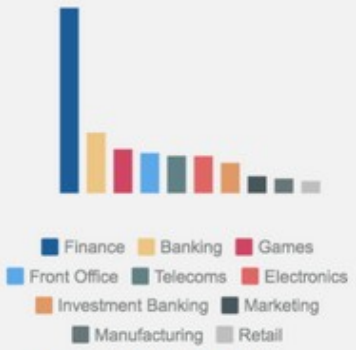
C++ is relatively more popular than other languages and technologies in Russia, Czech Republic, Hungary, France, Singapore, Finland, Israel and Germany.

University of Essex

Source: https://www.i-programmer.info/news/184-cc/8874-all-about-cc-developers.html

All about C++

#5



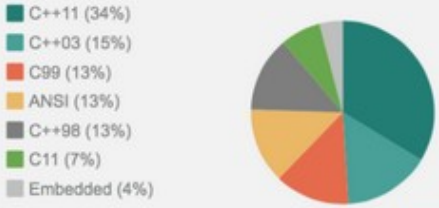
Industry distribution

Based on the C/C++ job ads we analyzed, C++ is most used in industries like Finance and Banking.

#6

Languages used with C++

Judging from job ads, the languages that are most typically used together with C++ include Java, C, C#, Python, SQL and JavaScript.

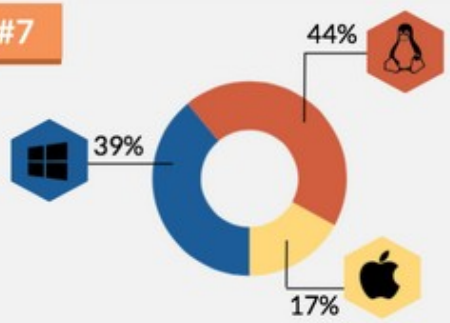


#8

C++ versions

The most popular C++ version is currently C++11, with a share of 34%.

#7



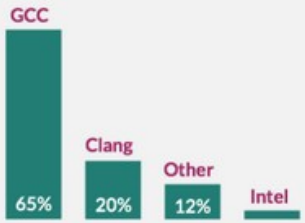
C++ developers by platform

The most popular platform is Linux, used by 44% of C/C++ developers, followed by Windows (39%) and OS X (17%).

University of Essex

All about C++

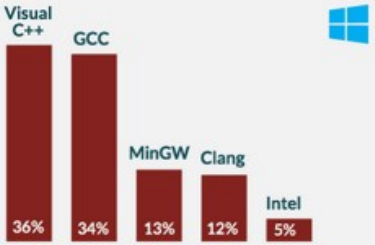
#9



Popular C++ compilers

GCC is by far the most popular compiler with 65%, followed by Clang with 20%. Together they cover 85% of all C++ developers.

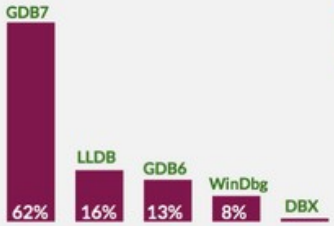
#10



C++ compilers on Windows

The title of the most popular C++ compiler on Windows is a virtual tie between Visual C++ (36%) and GCC (34%). MinGW (with no exact compiler name) and Clang are also tied with ~12% each. Intel comes in last at 5%.

#11

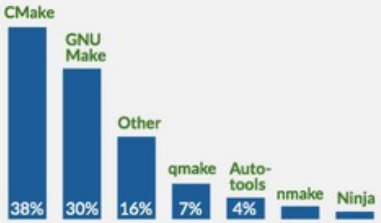


Popular C++ debuggers

The runaway leader among debuggers used by C++ developers is GDB7 (62%), with LLDB and GDB6 trailing with 16% and 13%, respectively.

The distribution on OS X is noticeably different: LLDB is ahead at 39% with GDB7 a close second at 32%.

#12



Popular C++ build systems

CMake and GNU Make build systems are the two close leaders, together accounting for 68% of all C++ developers.

#13

Compiler + Build tool + Debugger

GCC + CMake + GDB7 toolchain takes the top spot for all C++ developers.

On OS X, however, that honor goes to Clang + CMake + LLDB.

#14

Sources used:

1. Our user Survey
2. Stackoverflow
3. Job ads: Indeed.com
4. TIOBE index
5. GitHub
6. Google Trends
7. Reddit
8. External reports

sex

Learning Outcomes

After completing this module, students will be expected to be able to

- explain the basic concepts and features of C++.
- describe the underlying memory model and explain the role of the
- execution stack and the heap.
- write object-oriented programs that incorporate basic C++ features such as pointers, references, inheritance, function overriding, operator overloading, exceptions, etc.
- make effective use of the C++ Standard Template Library.

Lecture Outline

The lectures for this module are divided into three main parts:

- Basics: fundamental types and variables, memory management, references, pointers, arrays, control structures, functions, classes and objects, operator overloading, an Array class, the string class, file processing, comparison of C++ and Java.
- Libraries: templates (function templates and class templates), containers, iterators, algorithms, the Standard Template Library (STL).
- Advanced topics: inheritance, polymorphism, exception handling.

Moodle

University of Essex

CONTACT YOUR LECTURER

m.barros@essex.ac.uk

University of Essex