Fpt University
TRƯỜNG ĐẠI HỌC FPT

Programming Fundamentals using C

Contents

- Why should you study this course?
- Prerequisites
- Course Objectives
- Course Description and Course Plan
- Materials/ References/ Tools
- Course Rules
- Evaluation Strategy
- How to study
- Academic policy
- Install tools for programming

Why should you study this course?

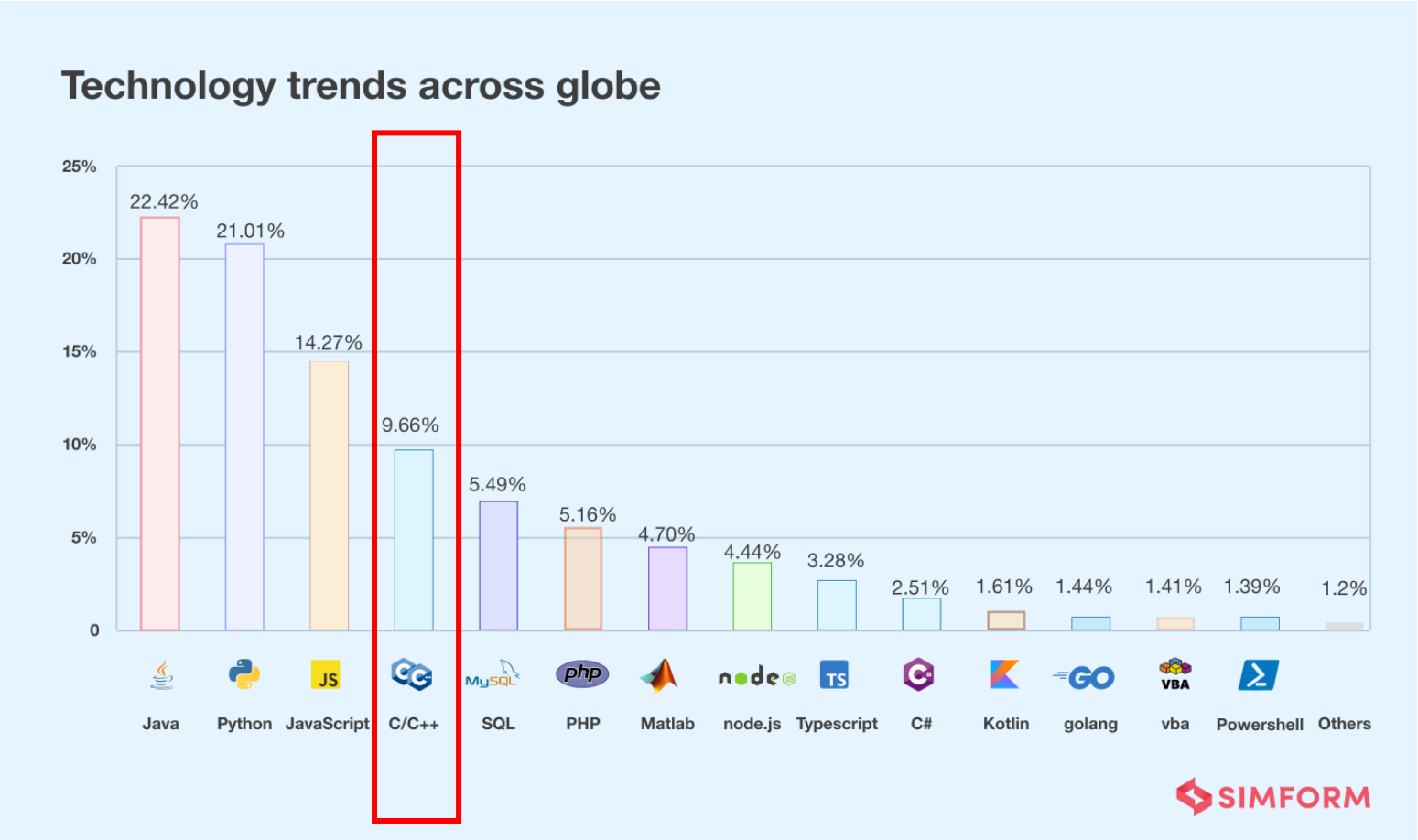
- **Foundation of Programming**: C is often considered the "**mother**" of modern programming languages. Many languages, like C++, Java, C# and Python, have roots in C, so learning it provides a solid foundation.
- **Performance**: C is a low-level language that allows for fine control over system resources and memory management, leading to high-performance applications.
- **System-Level Programming**: C is widely used in system programming, including operating systems and embedded systems.

Why should you study this course?

- **Rich Library Support**: C has a vast standard library, providing numerous built-in functions that can simplify development tasks.
- **Problem-Solving Skills**: Learning C enhances your problem-solving skills as it often requires a deeper understanding of algorithms and data structures.
- **Community and Resources**: C has a long-standing community, offering a wealth of resources, tutorials, and forums to help you learn and fix bug source code.

Why C is chosen?

- Top ten common programming languages:



Prerequisites

- Requires knowledge such as:
  - English reading comprehension
  - Basic logical thinking in mathematics

Course Objectives

- Explain the way to solve a real problem using computer.
- Understand the basic concepts computer system, and software development.
- Understand the basic concepts of programming, focus on procedure programming, testing and debugging, unit testing.
- Read and understand the simple C programs.
- Solve real problems using C.
- Individual and team work behaviors.

Course Description

1. Introduction to PFC
2. Basic Computation
3. Basic Logic constructs
4. Modules and Functions
5. Pointers
6. Libraries
7. Contiguous Storage using Arrays and Struct
8. Strings
9. Working to file: Text and Binary files

Course Plan

See course plan on FLM

Materials/ References

1) **Foundations of Programming Using C** (Hard Copy - FPT University Library)

   Author: Evan Weaver

   Published Date: July, 2006

2) **The C Programming Language**

   Authors: Brian W. Kernighan, Dennis M. Ritchie

   Published Date: 1988; Edition: 2nd Edtitions

3) **MOOC: Introduction to C Course**

   Link: https://intro2c.sdds.ca/

   Authors: Chris Szalwinsk and Seneca College

Tools

- Tool (IDE): **Dev-C++ version 6.3 (TDM-GCC 9.2)**
- Link download: https://www.embarcadero.com/free-tools/dev-cpp

Course Rules

- **How to conduct**
  - Prepare contents of the next session at home
  - Following lessons in classroom
  - Completing chapter assessments in time and Quizzes (https://exam.fpt.edu.vn)
  - ***Write reports*** of all labs and assignments to your notebook
- **Communication**
  - Class
  - Interchange by CMS, EduNext
  - Discussing actively in your team and classroom
  - Free to question and answer
- **Others**
  - Off phone, no game, no chat in class
  - Use laptop under teacher's instruction

Evaluation Strategy

- **Must attend more than 80% of contact hours (if not, not allow to take exam)**
- **Evaluating**
  - Maximum score: 10
  - On-going assessments:
    - 02 Progress Test (PT, 15%)
    - 05 Workshops (WS, 10%)
    - 01 Assignment (AS, 15%)
    - 01 Practical Exam (PE, 30%)
  - Final Exam (FE, 30%)
  - Total score=0.15*PT + 0.1*WS + 0.15*AS + 0.3*PE + 0.3*FE
- **Pass:**
  - Every on-going assessments component >0
  - Final Exam Score >=4 and Final Result >=5
- **Final Exam retake only when not passed**

How to study

- This course is complex knowledge (however, it's attractive and exciting), so you need to keep a tight grip on it
  - **Read**
    - On the books to get the general concept
    - Reference, study, collection from anywhere else (internet, your classmate, forum …)
  - **Attend lectures**
    - Listen, understand, then make your notes
    - Give your explanation about some topic in lectures
    - Ask questions
    - Give some examples that do not exist in your book
    - Practice all the exercises, demo to make your sense
  - **After classes**
    - Discuss your classmate indirectly, on the forum
    - Analyze, design, and implement workshops and assignments. **Write reports** in your notebook
    - Build your team in yourselves to support together in studying

Academic policy

- Cheating, plagiarism and breach of copyright are serious offenses under this Policy.
  - **Cheating**
    - Cheating during a test or exam is construed as talking, peeking at another student's paper or any other clandestine method of transmitting information.
  - **Plagiarism**
    - Plagiarism is using the work of others without citing it; that is, holding the work of others out as your own work.
  - **Breach of Copyright**
    - If you photocopy a textbook without the copyright holder's permission, you violate copyright law.

Enjoy the Course

- Be enthusiastic about the material because it is interesting, useful and an important part of your training as a software engineer.
- Our job is to help you learn and enjoy the experience.
- We will do our best but we need your help.
- So let's all have fun together with Programming Fundamentals using C language !!!

Install tools for programming

Q&A