CSE 124 Course Plan — Programming with C and C++

Nov 25, 2026 · @ATIK

Course information

The course runs as 42 classes of 50 minutes (14 weeks × 3 classes), which matches the 42 contact hours in the syllabus. Weeks 1–5 cover Problem Solving Techniques (15 hours) and Weeks 6–14 cover Structured Programming Language (27 hours).

Item	Detail
Course	CSE 124, Programming with C and C++ (ISCED 0741-124)
Type / status	GED, Theory
Credit / marks	3.0 credits, 100 marks
Prerequisite	Fundamentals of Computer
Schedule	14 weeks × 3 classes × 50 min = 42 classes

Part	Syllabus content	Classes	Weeks	CLOs
A	Problem Solving Techniques	15	1–5	CLO1, CLO2, CLO3, CLO4
B	Structured Programming Language (C)	27	6–14	CLO1, CLO2, CLO3, CLO4

CLO1: problem-solving with algorithms. CLO2: constants, variables, expressions, data types, operators. CLO3: decisions, loops, functions, arrays, strings, pointers. CLO4: unions, structures, basic data files.

Part A — Problem Solving Techniques (Weeks 1–5, Classes 1–15)

Students learn to analyse a problem and write an algorithm and a flowchart for it before writing any C. The part ends with coding, debugging and documentation, which leads into Part B.

Week	Class	Topic and content	CLO
1	1	Course introduction: syllabus, assessment, how computers solve problems, the program development life cycle	CLO1
1	2	Problem analysis: understanding the problem, identifying input, process and output, stating assumptions	CLO1
1	3	Algorithms: definition, properties (finiteness, definiteness, input, output, effectiveness), writing pseudocode	CLO1
2	4	Sequential algorithms: sum and average, area and perimeter, temperature conversion, swapping two values	CLO1, CLO2
2	5	Algorithms with decisions: even or odd, largest of three numbers, leap year, grade from marks	CLO1, CLO3
2	6	Algorithms with repetition: sum of N numbers, factorial, GCD, checking a prime number	CLO1, CLO3
3	7	Flowcharts: standard symbols, drawing rules, flowcharts for sequential problems	CLO1
3	8	Flowcharts with decisions: two-way and multi-way branching	CLO1, CLO3
3	9	Flowcharts with loops; converting between an algorithm and a flowchart	CLO1, CLO3
4	10	Problem-solving strategies: top-down design, stepwise refinement, modular decomposition	CLO1
4	11	Tracing and dry runs: trace tables, checking an algorithm by hand, a first look at efficiency	CLO1
4	12	Algorithms on lists of data: finding maximum and minimum, counting, linear search	CLO1, CLO3
5	13	Coding: from algorithm to program; compiler, interpreter, linker and IDE; structure of a first C program	CLO1
5	14	Debugging: syntax, runtime and logic errors; reading compiler messages; testing with test cases; using a debugger	CLO1

Week	Class	Topic and content	CLO
5	15	Documentation: comments, naming, indentation and coding style; internal and external documentation. **Quiz 1** (20 min)	CLO1

Part B — Structured Programming Language (Weeks 6–14, Classes 16–42)

This part follows the syllabus order from C basics to files, then finishes with command-line arguments, error handling, graphics and libraries. Each class should include at least one complete program written live in class.

Week	Class	Topic and content	CLO
6	16	Overview of C: history and features, structure of a C program, the compile process (preprocess, compile, assemble, link)	CLO2
6	17	Character set, tokens, keywords, identifiers; constants and variables; declaring and initialising variables	CLO2
6	18	Data types: int, char, float, double; modifiers; sizes and ranges; sizeof; implicit and explicit type conversion	CLO2
7	19	Operators I: arithmetic, relational, logical, assignment, increment and decrement	CLO2
7	20	Operators II: bitwise, conditional and comma; precedence and associativity; evaluating expressions	CLO2
7	21	Input and output: standard input and output, printf and scanf, format specifiers, getchar and putchar, formatted I/O	CLO2
8	22	Control structures I: if, if-else, nested if, else-if ladder	CLO3
8	23	Control structures II: switch, break, goto; the conditional operator compared with if-else	CLO3
8	24	Loops I: while and do-while; entry-controlled and exit-controlled loops	CLO3

Week	Class	Topic and content	CLO
9	25	Loops II: for loop, nested loops, pattern printing	CLO3
9	26	break and continue; loop problems: prime numbers, Fibonacci series, reversing and summing digits	CLO1, CLO3
9	27	Functions: definition, prototype, calling, return values; program structure with functions	CLO3
10	28	Parameter passing conventions: call by value and call by reference (with a first look at pointers)	CLO3
10	29	Scope rules and storage classes: auto, extern, static, register; local and global variables	CLO3
10	30	Recursion: factorial, Fibonacci, GCD, Tower of Hanoi; recursion compared with iteration	CLO1, CLO3
11	31	Header files and the preprocessor: #include, #define, macros, conditional compilation; writing your own header file	CLO3
11	32	One-dimensional arrays: declaration, initialisation, traversal, linear search, bubble sort	CLO3
11	33	Multidimensional arrays: matrix addition, transpose and multiplication; passing arrays to functions	CLO3
12	34	Pointers I: addresses, declaration, dereferencing, pointer arithmetic	CLO3
12	35	Pointers II: pointers and arrays, pointers and functions, dynamic memory with malloc, calloc and free	CLO3
12	36	Strings: character arrays, string.h functions, strings with pointers, arrays of strings	CLO3
13	37	User-defined data types I: structures, accessing members, arrays of structures, nested structures, pointers to structures	CLO4
13	38	User-defined data types II: unions, enumerations, typedef; structure compared with union	CLO4
13	39	File access: fopen and fclose, file modes, fprintf and fscanf, fgets and fputs, text and binary files	CLO4

Week	Class	Topic and content	CLO
14	40	Variable length argument lists (stdarg.h); command line parameters (argc and argv)	CLO4
14	41	Error handling: return codes, errno and perror, feof and ferror, exit. Introduction to graphics (graphics.h basics)	CLO4
14	42	Library functions (math.h, stdlib.h, ctype.h, time.h); linking with libraries; course review and final exam guidance	CLO4

Assessment schedule

Assessments follow the syllabus strategy for each CLO. The midterm and final are held outside the 42 lecture slots, and each quiz takes 20 minutes of a class.

When	Assessment	Coverage	CLO	Type
Week 5, Class 15	Quiz 1	Algorithms, flowcharts, tracing (Classes 1–12)	CLO1	Formative
Weeks 2–5	Homework sets	Algorithm and flowchart exercises	CLO1	Formative
After Week 8	Midterm exam	Classes 1–24: problem solving, C basics, operators, I/O, decisions, while loops	CLO2, CLO3	Summative
Set Week 10, due Week 12	Programming assignment	Loops, functions, recursion, arrays and strings	CLO3	Formative
Week 12, Class 36	Quiz 2	Arrays and pointers (Classes 32–35)	CLO3	Formative
After Week 14	Final exam	Whole course, with weight on Classes 25–42	CLO3, CLO4	Summative

The marks split between class tests, attendance, midterm and final is not in the syllabus pages provided, so set it to the department's rule.

Reading guide and notes

Part	Main text	Supporting texts
A: Problem Solving (Weeks 1–5)	Dromey, *How to Solve It by Computer*	Balagurusamy, *Programming in ANSI C* (introductory chapters)
B: C language (Weeks 6–14)	Balagurusamy, *Programming in ANSI C*	Kernighan & Ritchie, *The C Programming Language*; King, *C Programming: A Modern Approach*; Kochan, *Programming in C*; Schildt, *Teach Yourself C*

- The course title includes C++, but the syllabus content lists only C topics. This plan follows the listed content. If the department expects C++, Classes 40–42 could become a short C++ introduction (classes, objects, cin and cout).
- Graphics (Class 41) depends on the compiler. graphics.h works only with Turbo C or WinBGIm setups, so check what the lab machines support before that class.
- CSE 124 is a theory course. If students have no linked lab, give a short take-home program each week from Week 6 so they practise coding.