

Chapter 3 – The C Integrated Development Environment (IDE)

✔ Short Questions with Easy Answers

Q1. What is Turbo C++?

👉 Turbo C++ is an **IDE (Integrated Development Environment)** for writing C/C++ programs, developed by **Borland International**.

- It is used to **create, edit, save, compile, debug, and run** C programs.
- Example: You write code → press F9 to compile → run it inside Turbo C++.

Q2. What are necessary steps to prepare a C program?

👉 Steps:

1. Create & Edit program
2. Save program
3. Compile (check for errors)
4. Link (combine code with libraries)
5. Load into memory
6. Run program

Example: Writing a program to add two numbers requires all these steps.

Q3. What are header files?

👉 Header files contain **built-in functions** grouped by purpose.

- Example:
 - #include <stdio.h> → for input/output (printf, scanf).
 - #include <math.h> → for math functions (sqrt, pow).

Q4. What is a C statement?

👉 Every instruction in C is a **statement** and ends with **semicolon (;)**.

- Example:

```
int sum = a + b;
```

Q5. What are syntax errors?

👉 Errors when rules of C language (syntax) are not followed.

- Detected by compiler.
- Example:

```
printf("Hello") // ❌ error (missing ;)
```

Q6. What are logical errors?

👉 Errors due to **wrong logic**, not wrong syntax.

- Not detected by compiler.

- Example: Writing `sum = a - b;` instead of `sum = a + b;`;

Q7. What are runtime errors?

👉 Errors that happen while **program is running**. Execution stops with an error message.

- Example: Dividing by zero $\rightarrow x = 5 / 0;$

Q8. What is ANSI C?

👉 The **standard version of C** developed by American National Standard Institute (ANSI) in the 1980s.

- It made C portable and uniform across all computers.

Q9. List any four advantages of C language.

1. Easy to learn.
2. Machine independent.
3. Standard syntax.
4. Shorter and efficient programs.

Q10. What is meant by machine independence?

👉 A program in **high-level language (like C)** can run on any computer (machine independent).

👉 Low-level languages (like Assembly) only run on specific computers (machine dependent).

✅ Long Questions (Easy Style with Examples)

Q1. Write the steps to develop a program in C.

1. **Analyze problem** \rightarrow Understand what to solve.
2. **Design solution** \rightarrow Flowchart/Algorithm.
3. **Code** \rightarrow Write program in C.
4. **Compile & Debug** \rightarrow Check errors.
5. **Run & Test** \rightarrow Try different inputs.
6. **Documentation** \rightarrow Explain code.
7. **Implementation & Maintenance** \rightarrow Final program delivered.

Q2. Describe the structure of a C Program.

👉 Basic structure:

```
#include <stdio.h> // Header file
```

```
int main() { // Main function
```

```
int a, b, sum; // Declarations
```

```
a = 5; b = 10; // Statements
```

```

sum = a + b;

printf("Sum = %d", sum); // Output

return 0;    // End program
}

```

Parts:

1. Header files
2. Main function
3. Variable declaration
4. Statements
5. Output
6. End

Q3. Describe Debugging feature in detail.

👉 **Debugging = finding and fixing errors.**

- IDE like Turbo C++ has built-in debugger.
- Features:
 - **Breakpoints** → stop at certain line.
 - **Step execution** → run line by line.
 - **Error messages** → show where problem is.
 - 👉 Example: Debugging a wrong formula in a student marks program.

Q4. Describe Integrated Development Environment (IDE).

👉 IDE = **software package** that provides everything in one place to write and run programs.

- Features of IDE:
 - Editor (write code)
 - Compiler (check errors)
 - Linker (connect with libraries)
 - Debugger (fix errors)
 - Runner (execute program)
- 👉 Example: Turbo C++, Code::Blocks, Dev C++, Visual Studio.

Q5. What is the difference between Compiler and Interpreter?

Compiler

Translates the **whole program** into machine code at once. Translates and runs program **line by line**.

Fast execution after compilation.

Example: C, C++ use compilers.

Interpreter

Slower because it runs line by line.

Example: Python, JavaScript use interpreters.