Lecture : 3 Conditional Statements & Loops

Key Focus :

if , else-if , else structure + character case detection in C++

Grading System Example :

Problem :

Assign grades based on marks :

- ≥90: **A**

- 80-89: **B**

- <80: **C**

Core Logic Flow :

```
marks ≥ 90? → Print "A" ✓
     ↓ No
80 ≤ marks < 90? → Print "B" ✓
     ↓ No
Print "C"
```

C++ code Structure :

```
if (marks >= 90) {
    cout << "A";
}
else if (marks >= 80 && marks < 90) {  // && = AND operator  【1017.44, type: source】
    cout << "B";
}
else {
    cout << "C";
}
```

Test Cases:

- marks = 95 → **A**

- marks = 82 → **B**

- marks = 75 → **C**

Rules:

- `if` : Always required

- `else if` : Optional, multiple allowed

- `else` : Optional, only once at end

Character Case Detection :

Problem :

Check if character is lowercase (a-z) or uppercase(A-Z)

```
char ch;
cin >> ch;

if (ch >= 'a' && ch <= 'z') {  // a-z range  【1200.159, type: source】
    cout << "Lowercase";
}
else {
    cout << "Uppercase";  // Guaranteed A-Z  【1238.159, type: source】
}
```

Why it works: Characters have sequential **ASCII values** internally

Method 2 : ASCII Numeric Values :

ASCII Ranges :

```
Uppercase A-Z: 65-90
Lowercase a-z: 97-122
```

```
if (ch >= 65 && ch <= 90) {  // Uppercase  【1387.559, type: source】
    cout << "Uppercase";
}
else {
    cout << "Lowercase";
}
```

Key Concept: **Implicit Type Conversion** - Compiler converts `char` to its **ASCII number** for comparison

Test Cases:

- ch = 't' → **Lowercase**

- ch = 'T' → **Uppercase**

Essential Syntax Rules :

Components	Usage Rules	
if	always 1st mandatory	if (condition) { }
else if	after if, multiple OK, checks only if previous false	else if (condition) { }
else	last only, 1 max, catches all remaining cases	else { }

Logical Operator Used :

&& (AND) : Both conditions must be true

```
marks >= 80 && marks < 90  // Range check
ch >= 'a' && ch <= 'z'     // Range check
```

Homework Problems :

Problem 1: Sum of multiples of 3

Input: n = 10

Numbers divisible by 3: 3, 6, 9

Output: 18

Approach:

1. Use **for/while loop** from 1 to n

2. Check `if(i % 3 == 0)`

3. Add to sum