Lecture 5 & Lecture 6 Notes (Loops, Math & Patterns in C++)

🔹 Lecture 5 – For Loop, Numbers & Maths Logic

1. Recap of Loops

Define & Simplify
Loop = ek machine jo ek kaam repeat karte hai bina thake.
Real life analogy → Agar tumhe 100 baar “Hello” bolna ho, tum manual karoge ya ek loop ko bol doge "repeat 100 times"?

Break into Fundamentals

- 3 parts of `for` loop → Initialization, Condition, Update.
- Flow → Start → Check condition → Run → Update → Repeat until condition false.

Rebuild Understanding
Example:

```
for(int i=1; i<=5; i++){
    cout << i << " ";
}
```

Output → `1 2 3 4 5`

---

2. Print Numbers & Alphabets

- Print numbers from **101 to 200**
- Print alphabets from **'a' to 'z'**
- Print numbers in **reverse order**

👉 Concept: Loops can increase (i++), decrease (i--), or even jump (i+=3).

---

3. Difference-based Printing

Print numbers from 1 to 100 but with step = 3.

```
for(int i=1; i<=100; i+=3){
    cout << i << " ";
}
```

Logic → Update part is flexible (not always `i++`).

---

4. Multiplication Table

```
int n=5;
for(int i=1; i<=10; i++){
    cout << n << " x " << i << " = " << n*i << endl;
}
```

---

5. Power of a Number

Idea → Multiply the base with itself `exp` times.

```
int base=2, exp=5, result=1;
for(int i=1; i<=exp; i++){
    result *= base;
}
```

---

6. Sum Calculations

- Sum of `n` natural numbers
- Sum of squares of `n` natural numbers

Formula help (shortcut math)

- Sum of n numbers = `n*(n+1)/2`
- Sum of squares = `n*(n+1)*(2n+1)/6`

---

7. Factorial of a Number

Factorial = product of numbers from 1 to n.

```
int fact=1;
for(int i=1; i<=n; i++){
    fact *= i;
}
```

---

8. Prime Number Check

Logic → Prime = divisible only by 1 & itself.

```
bool isPrime=true;
for(int i=2; i<=n/2; i++){
    if(n%i==0) { isPrime=false; break; }
}
```

---

9. Fibonacci Series

Logic → Next = Previous + Previous-Previous.
Start with 0,1 → 0,1,1,2,3,5,8…

🔹 Lecture 6 – Nested Loops & Patterns

1. Concept of Nested Loops

Loop inside another loop = Nested Loop.
Real life analogy → School → Each class has students. Outer loop = class, inner loop = students.

---

2. Solid Rectangle (Stars)

```
for(int i=1; i<=3; i++){       // rows
    for(int j=1; j<=5; j++){   // cols
        cout << "* ";
    }
    cout << endl;
}
```

Output → rectangle of 3x5 stars.

---

3. Number Patterns

- Print numbers in sequence
- Descending order
- Ascending square patterns

---

4. Alphabet Patterns

- Print alphabets like a rectangle or triangle
- Use ASCII values → `'A' = 65`, `'a' = 97`.

Example:

```
for(char c='A'; c<='E'; c++){
    for(int j=1; j<=5; j++){
        cout << c << " ";
    }
    cout << endl;
}
```

---

5. Debugging & Variations

Patterns = visual way of logic building.

- Change rows = outer loop.
- Change columns = inner loop.
- Use ASCII addition to print A, B, C etc.

🔥 Test & Iterate (Practice Ideas)

1. Print numbers from 50 to 1.
2. Multiplication table of 19.
3. Sum of cubes of first n numbers.
4. Solid square of 6x6 stars.
5. Pattern → Right triangle of numbers.
6. Pattern → Alphabets in descending order.

👉 **Bottom Line**:

- Lecture 5 → Loops + Maths logic building.
- Lecture 6 → Nested loops + Patterns (visual imagination).
