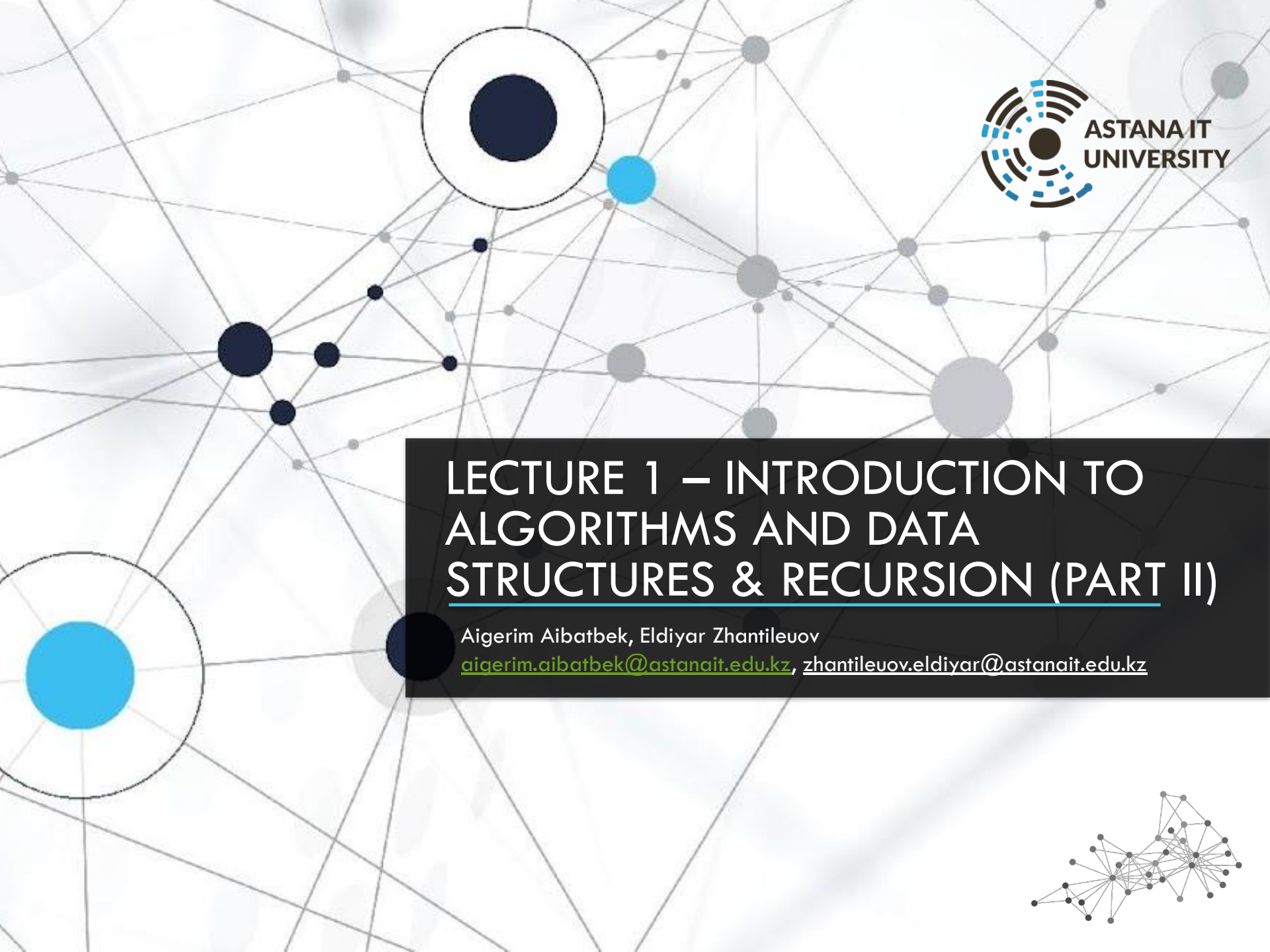
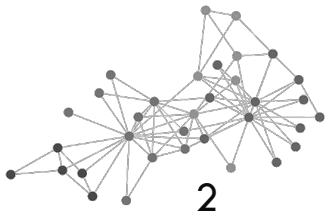


LECTURE 1 – INTRODUCTION TO ALGORITHMS AND DATA STRUCTURES & RECURSION (PART II)

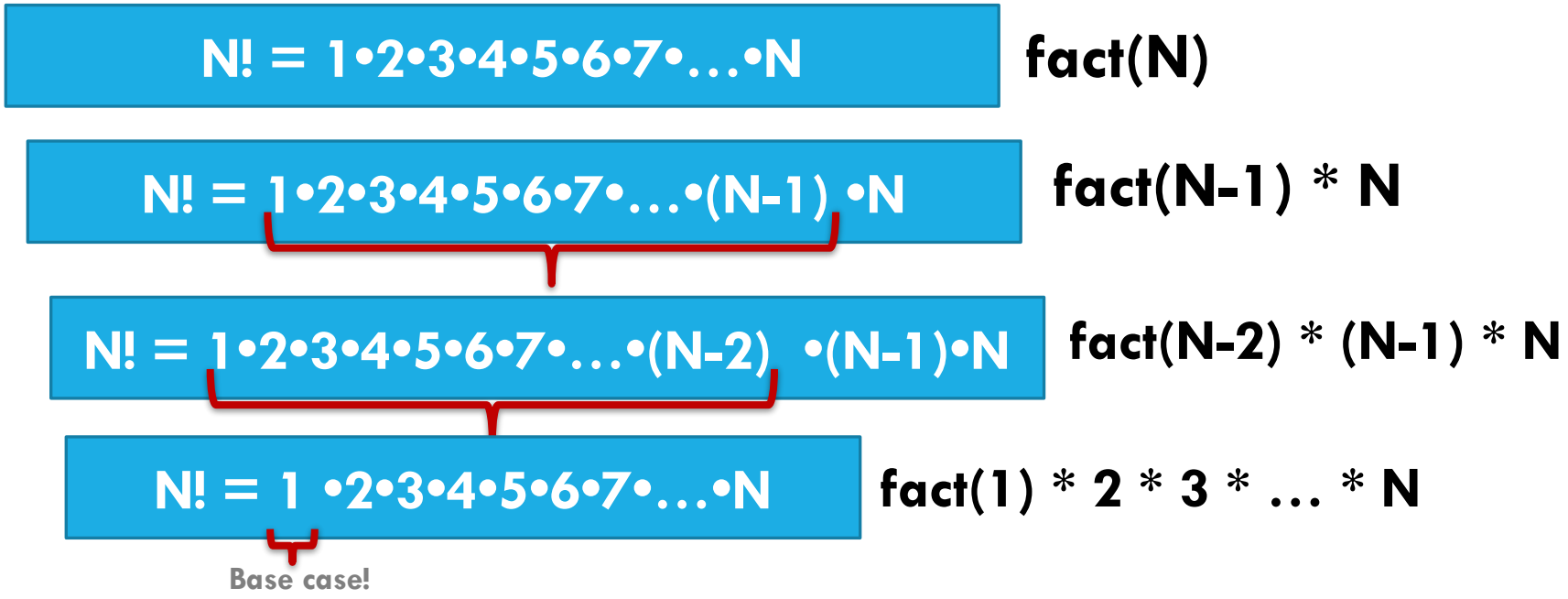
Aigerim Aibatbek, Eldiyar Zhantileuov
aigerim.aibatbek@astanait.edu.kz, zhantileuov.eldiyar@astanait.edu.kz

CONTENT

1. Example 2 – Factorial of the Number
2. Iteration vs Recursion
3. How to create a recursive algorithm?



EXAMPLE 2 – FACTORIAL OF THE NUMBER



EXAMPLE 2 – FACTORIAL SOLUTION

ASTANA IT UNIVERSITY

```java
public static int factorial(int N) {
    if (N <= 1) return 1; // base case

    return factorial( N: N - 1) * N;

}

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    int n = sc.nextInt();

    int result = factorial(n);

    System.out.println(result);
}
```

```java
int factorial(int N) {
    if (N <= 1) return 1; // base case

    return factorial( N: N - 1) * N;

}

void main() {
    int result = factorial( N: 3);

    System.out.println(result);
}
```

Stack

```java
void main() {
    int result = factorial( N: 3);

    System.out.println(result);
}
```

ITERATION VS RECURSION

ASTANA IT UNIVERSITY

Repetition

- Iteration: explicit loop
- Recursion: repeated function calls

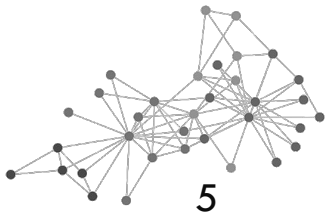
Termination

- Iteration: loop condition fails
- Recursion: base case recognized

Both can have infinite loops

Balance between performance (iteration) and good software engineering (recursion)

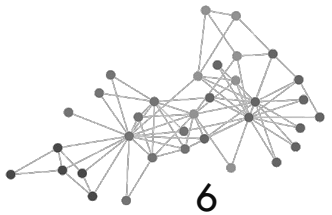
Criteria	Iteration	Recursion
Mode of implementation	Implemented using loops	Function calls itself
State	Defined by the control variable's value	Defined by the parameter values stored in stack
Progression	The value of control variable moves towards the value in condition	The function state converges towards the base case
Termination	Loop ends when control variable's value satisfies the condition	Recursion ends when base case becomes true
Code Size	Iterative code tends to be bigger in size	Recursion decreases the size of code
No Termination State	Infinite Loops uses CPU Cycles	Infinite Recursion may cause Stack Overflow error or it might crash the system
Execution	Execution is faster	Execution is slower



HOW TO CREATE A RECURSIVE ALGORITHM?

ASTANA IT UNIVERSITY

1. Think about a problem at a high level of abstraction
2. Figure out the base case for the program
3. Redefine the answer in terms of a simpler sub-problem
4. Combine the results in the formulation of the answer



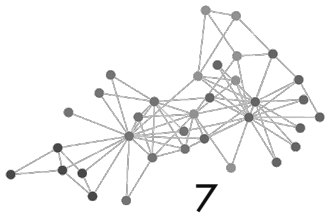
LITERATURE

Algorithms, 4th Edition, by Robert Sedgewick and Kevin Wayne, Addison-Wesley

- Chapter 1.1

Grokking Algorithms, by Aditya Y. Bhargava, Manning

- Chapter 3



ASTANA IT UNIVERSITY

GOOD LUCK!