Advanced Data Structure and Algorithms fo Problem Solving

Lecture # 04

Vineet Gandhi

Center for Visual Information Technology (CVIT),

IIIT Hyderabad

Recursion

- Recursion is a mathematical technique that evaluates a function by calling the same function repeatedly on smaller inputs.
- Most programming languages support such a style of programming.
  - Often very elegant to study.
- Helps in problem solving too.

Recursion

- **Q:** How many twists does it take to screw in a light bulb?
  A: Is it already screwed in? Then zero. If not, then twist it once, ask me again, and add 1 to my answer.

Recursion

- Relates to mathematical induction
- Divide and Conquer algorithms

Lets start with an examples

- A mathematical view of computer science
- factorial(n) = n* factorial(n-1)

Three Laws of Recursion

- factorial(n) = n* factorial(n-1)
- A recursive algorithm must change its state and move toward the base case.
- A recursive algorithm must call itself, recursively.

Factorial – deeper look

Factorial – deeper look

- Space and time complexity

Inductive Reasoning

- A recursive algorithm must have a **base case**.
- If I call factorial(n) with n=1, I am done
- If I call factorial(n) with n>1, it makes a recursive call with a smaller value of n; must eventually reach n=1.

Recursion and Induction

Recursion with multiple base cases

Recursion with multiple base cases

Fibonacci deeper look

- Time and space complexity

Recursion is not always good

Recursion with memoization

Exponentiation

Modular Exponentiation

Thank You