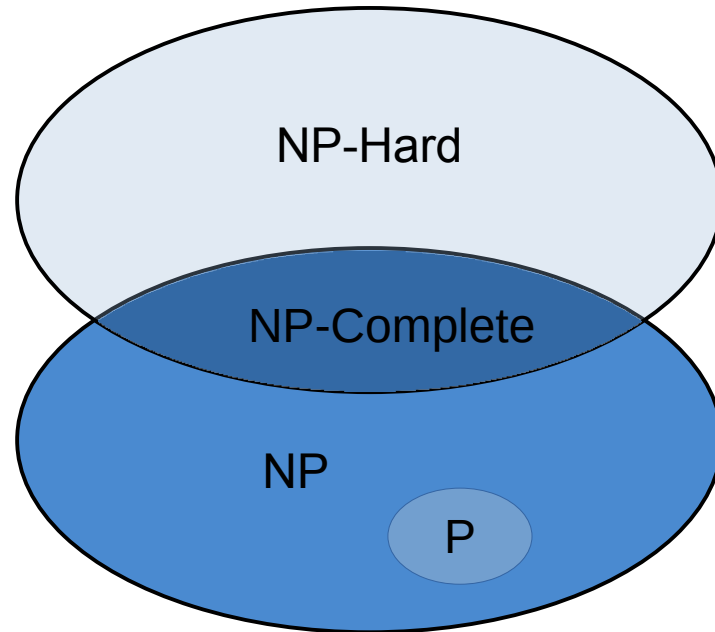


# Problems classification

Simeon Monov

# Problems Classification

- Easy problems (P)
- Medium problems (NP)
- Hard problems (NP-Complete)
- Hardest problems (NP-Hard)



# Problem complexity

Big-O values:

- $O(1)$  – constant time
- $O(\log_2(n))$  – logarithmic time
- $O(n)$  – linear time
- $O(n^2)$  – quadratic time
- $O(n^k)$  – polynomial time
- $O(k^n)$  – exponential time
- $O(n!)$  – factorial time

# Polynomial Algorithms

- Polynomial algorithms we can be solved in polynomial time, like logarithmic, linear or quadratic time. They all have complexity of some  $O(n^k)$ :
  - Mathematical operations: addition, subtraction, division, multiplication
  - Finding primes
  - Hashtable lookup, sorting, string operations
  - Linear or binary search algorithms

# NP Algorithms

- NP algorithms (Non-deterministic Polynomial algorithms) cannot be solved in polynomial time. However, they can be verified in polynomial time. These algorithms can be solved in exponential time  $O(k^n)$ :
  - Integer factorization

# NP-Complete Algorithms

- NP-Complete problems are problems for which no efficient solution algorithm has been found but they can be solved by a non-deterministic algorithm in polynomial time.
  - Traveling Salesman
  - Knapsack
  - Graph Coloring

# NP-Hard algorithms

- NP-Hard algorithms are the hardest, most complex problems in computer science. They are not only hard to solve but are hard to verify as well:
  - K-means Clustering
  - Traveling Salesman Problem
  - Graph Coloring