

## Summary: Lecture 2

Summary for the chapters 7.1 *Complexity classes* and 7.2 *Hierarchy problem*. [2]

### Complexity classes

#### Background knowledge:

A complexity class is a set which contains problems with similar complexities. The complexities are examined in regards of a specific resource, for example time or space. For the problems the most efficient solution/algorithm is analysed.

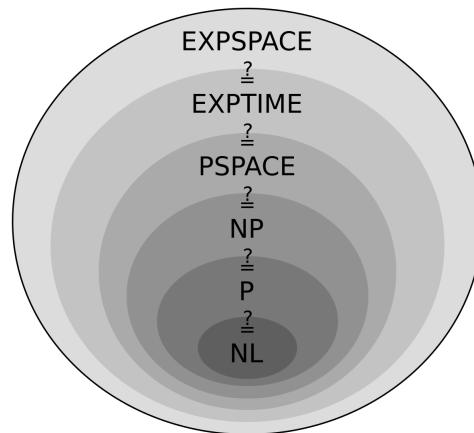


Figure 1: Complexity classes [1]

Usually the complexity depends on the input size. With the asymptotic complexity, classes are build, which are the complexity classes. [3]

#### Summary:

##### Parameters of complexity classes:

- **Model of computation:**  
here: multistring Turing Machine
- **Mode of computation:**  
for example: deterministic or non-deterministic (deterministic: the computer will always produce the same output for a given input while going through the same states, non-deterministic: can show different behaviors for the same input)
- **Ressources:**  
something *expensive* that the machine uses up, for example: time or space
- **Restrictions/Bound:**  
for example: upper bound, lower bound as a function  $f : \mathbb{N} \rightarrow \mathbb{N}$

### Hierarchy problem

## References

- [1] *Complexity classes diagram image source*. [https://en.wikipedia.org/wiki/Complexity\\_class](https://en.wikipedia.org/wiki/Complexity_class).
- [2] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley Publishing Company, 1994.
- [3] Prof. Dr. Thomas Schwentick. *Lecture notes in Grundbegriffe der theoretischen Informatik*. [https://www.cs.tu-dortmund.de/nps/de/Studium/Ordnungen\\_Handbuecher\\_Beschluesse/Modulhandbuecher/Archiv/Bachelor\\_LA\\_GyGe\\_Inf\\_Modellv/\\_Module/INF-BfP-GTI/index.html](https://www.cs.tu-dortmund.de/nps/de/Studium/Ordnungen_Handbuecher_Beschluesse/Modulhandbuecher/Archiv/Bachelor_LA_GyGe_Inf_Modellv/_Module/INF-BfP-GTI/index.html).