Homework 2: Code Contracts
Task 1

• Implement data structure in C#
  ▪ Elements: integer type, duplicates allowed
  ▪ Access: using element index or actual value
  ▪ Operations
    • void Add(int val)
    • int Get(int index)
    • int GetHigher(int val)
      ▪ It should return the least element greater than val
    • void Remove(int index)
    • void RemoveAll(int val)
    • void Sort()
    • int FindMin()
    • bool Contains(int val)
    • void Clear()
    • int Size()
Task 2

- Define contracts for all operations provided by your data structure
  - Contracts should capture the expected behavior
    - All typical usage patterns supported by the operations
  - Try to cover also some important corner cases
    - Example: index out of bounds
Task 3

• Write small test client for the data structure
  ▪ It should exercise typical usage patterns and some important corners cases

• Note for tasks 1+3
  ▪ We will not judge the quality of your code in C#
    • Some prefer and use other languages (Java, C, C++, ...)
Task 4

• Use static checker to verify the implementation of your data structure against the contracts

• Use the runtime checker on your small test client

• Necessary software
  □ Visual Studio + plugins for Code Contracts
    • https://www.microsoft.com/en-us/research/project/code-contracts/
Task 5

- Document your solution
  - Informally describe what non-trivial properties you specified using Code Contracts
    - “why you did what you did”
  - Positive experience: what contracts (properties) you were able to successfully verify
  - Negative experience: what are the major observed limitations of Code Contracts
  - For each reported spurious error (if you get some), try to explain why the particular checker reported the error in your opinion
  - Also discuss missed errors (and possible reasons)
Alternative: Viper

- You are allowed to use Viper (language, tools) instead of C# and Code Contracts
  - [http://viper.ethz.ch/](http://viper.ethz.ch/)
  - Online web interface
  - Plugin for VS Code
  - No runtime checker

- Special task
  - Compare abilities of VC generator and symbolic execution

- Read the tutorial
  - [http://viper.ethz.ch/tutorial/](http://viper.ethz.ch/tutorial/)
Organization

- Deadline: **2.5.2019**

- Submission
  - E-mail: parizek@d3s.mff.cuni.cz