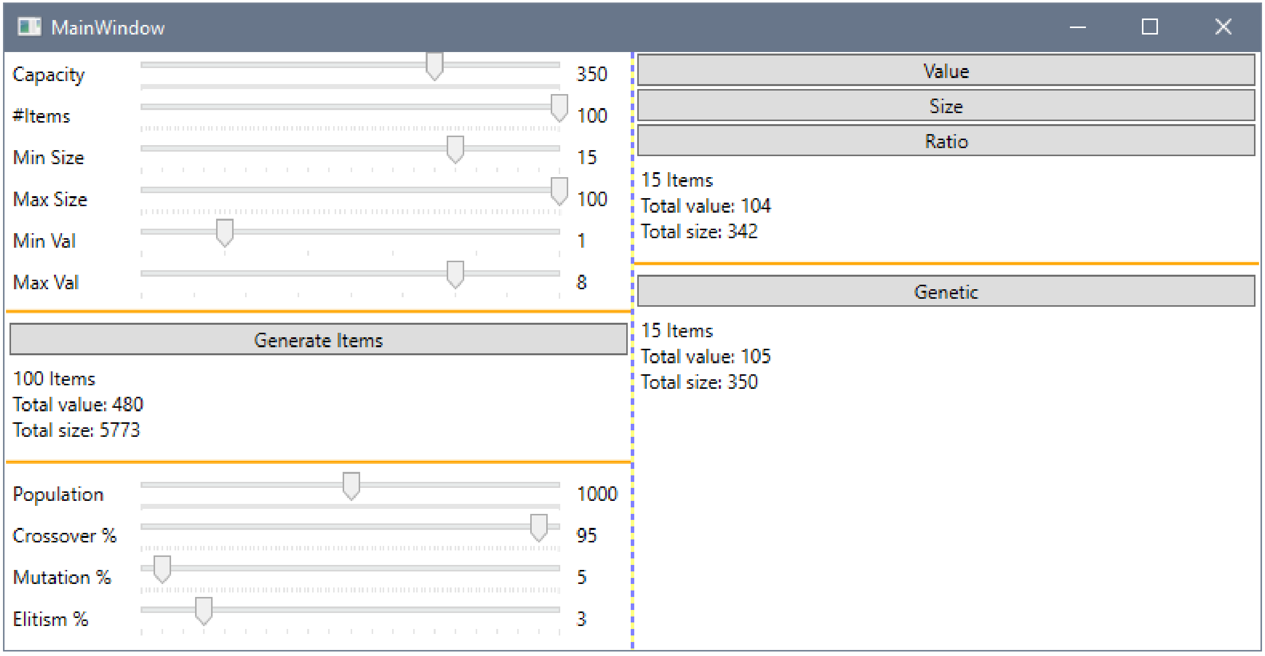
# Genetic Algorithm / Backpack problem

Implement the genetic algorithm as discussed in the presentation.



Capacity 350
#Items 100
Min Size 15
Max Size 100
Min Val 1
Max Val 8

100 Items
Total value: 480
Total size: 5773

Population 1000
Crossover % 95
Mutation % 5
Elitism % 3

15 Items
Total value: 104
Total size: 342

15 Items
Total value: 105
Total size: 350

Compare the result with the "ratio" variant, where the relatively most expensive objects are packed first.
Also experiments with the parameters of the algorithm (crossover, mutation, population, elitism).

The project with the UI is attached to this document.
Most of the helper methods are also already programmed, so you can focus on the implentation of the algorithm and how to handle crossover and mutation.
The following methods have to be implemented (they currently hold a compileable dummy implementation):

## Backpack.cs

- Mutate
- CrossOver
- Repair

## ConventionalAlgos.cs

- GetItemsBaseOnValue
- GetItemsBaseOnSize
- GetItemsBaseOnRatio
- GetItemsThatFit

## GeneticAlgorithmEngine

- section FindBest
  - FindBest
- section Next Generation
  - CreateNextGeneration
  - GetBestAndCalculateTotalFitness
  - AddElite
  - ShouldTerminate
- section Selection
  - SelectCandidateByRouletteWheel
  - SelectCandidateViaTournament