TECHNICKÁ UNIVERZITA V KOŠICIACH

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

GENETIC ALGORITHM

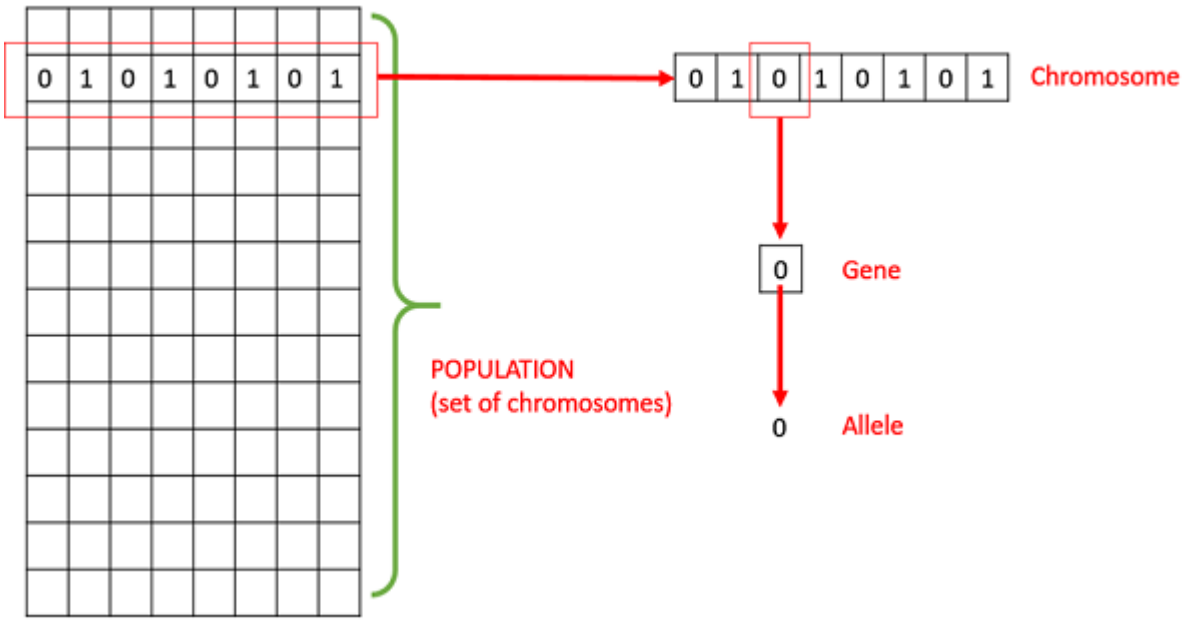
Olha Shvenda
Oleksandr Bodnar
Nikita Shalashkov
Vladyslav Kurtanidze

2022

CONTENT:

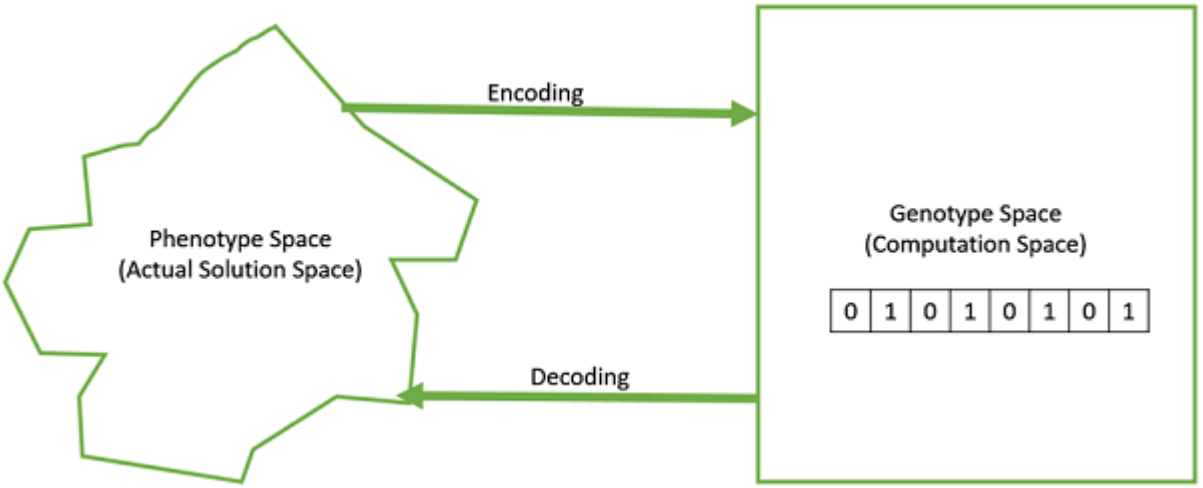
1. What are genetic algorithms?

These are adaptive heuristic algorithms that belong to evolutionary algorithms. They are based, as you already understood, on genetics and natural selection (survival of the fittest). Most often they are used for search problems and optimization problems.



Source: Tutorialspoint

- *Population* - it is a subset of all possible solutions
- *Chromosome* - one solution
- *Gene* - one element in chromosome
- *Allele* - the value a gene takes

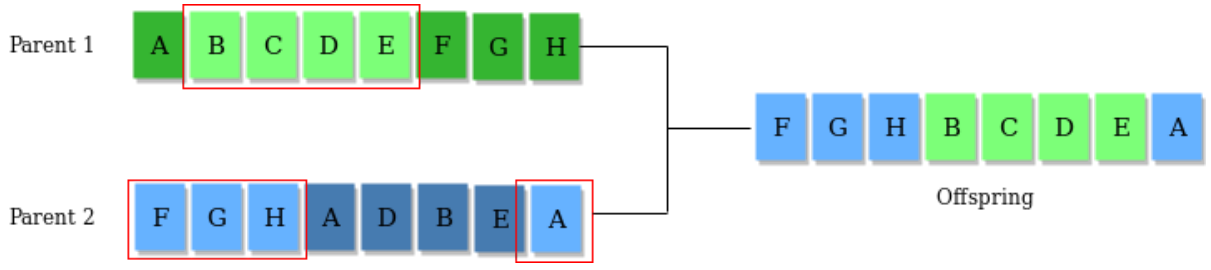


Source: Tutorialspoint

- *Genotype* - space, where solutions are represented in a way which can be easily understood
- *Phenotype* - real-world solution space
- *Fitness score* - amount of likes (the ability of an individual to “compete”)

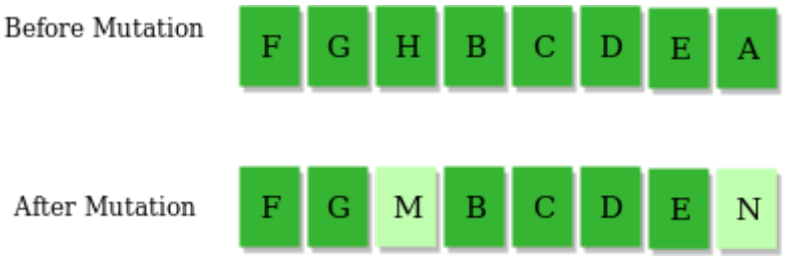
Operators of Genetic Algorithms

1. Selection Operator - more preference to the individuals with good fitness scores
2. Crossover operator - we select individuals by the highest fitness score using the selection operator. After that, we randomly select places for "mating" and exchange genes in these places. A new individual (offspring) is created.



Source: Geeksforgeeks

3. Mutation operator - insertion of random genes into offspring to maintain diversity



Source: Geeksforgeeks

2. Pseudocode of our algorithm

A description of the entire algorithm:

```
1) Randomly initialize populations p
2) Determine fitness of population
3) Until convergence repeat:
      a) Select parents from population
      b) Crossover and generate new population
      c) Perform mutation on new population
      d) Calculate fitness for new population
```

Pseudocode:

```
GA()
   initialize population
   find fitness of population

   while (termination criteria is reached) do
      parent selection
      crossover with probability pc
      mutation with probability pm
      decode and fitness calculation
      survivor selection
      find best
   return best
```

3. Problems that can be solved

- Traveling salesman problem
- Vehicle routing problem
- SEND+MORE=MONEY problem

- OneMax problem
- Our problem

4. Our example problem: knapsack problem

We tried to find the most real-life problem. This problem is considered the oldest and most investigated combinatorial search problems. So, imagine the situation: you need to pack a backpack to go to university/gym/trip. There are many things you would like to take but there is a weight limit.
In your mind, each item has a certain number of "likes" that show how much you want to take this thing with you or how badly you need it.
Also, each item has a certain size and weight. All these values will be taken into account when packing your backpack.

Components we have:
- things with "likes" and weight (fitness score)
- backpack with limited weight

The goal is to find a combination of things that will provide the maximum total value of likes but will not weigh more than the backpack limit.

Even with the *naive implementation* of just trying out every possible combination, we will have 32 different combinations when it comes to 5 items. My computer can do this in 0.000003 seconds. What will happen if we add more items?

ITEMS	COMBINATIONS	SECONDS
5	32	0.000003
10	1,024	0.000807
20	1,048,576	1.000000
21	2,097,152	1.940000
22	4,194,304	3.910000

Source: Youtube

5. Experiments and results

We have 4 .csv files, which represent 4 knapsack problems:

1. BoySchoolBackpack.csv
2. GirlScoolBackpack.csv
3. BoyTravelBackpack.csv
4. GirlTravelBackpack.csv

We will initialize problems with these parameters:

```
Max generation = 50
Population size = 500
Crossover = 0.9
Mutation = 0.1
```

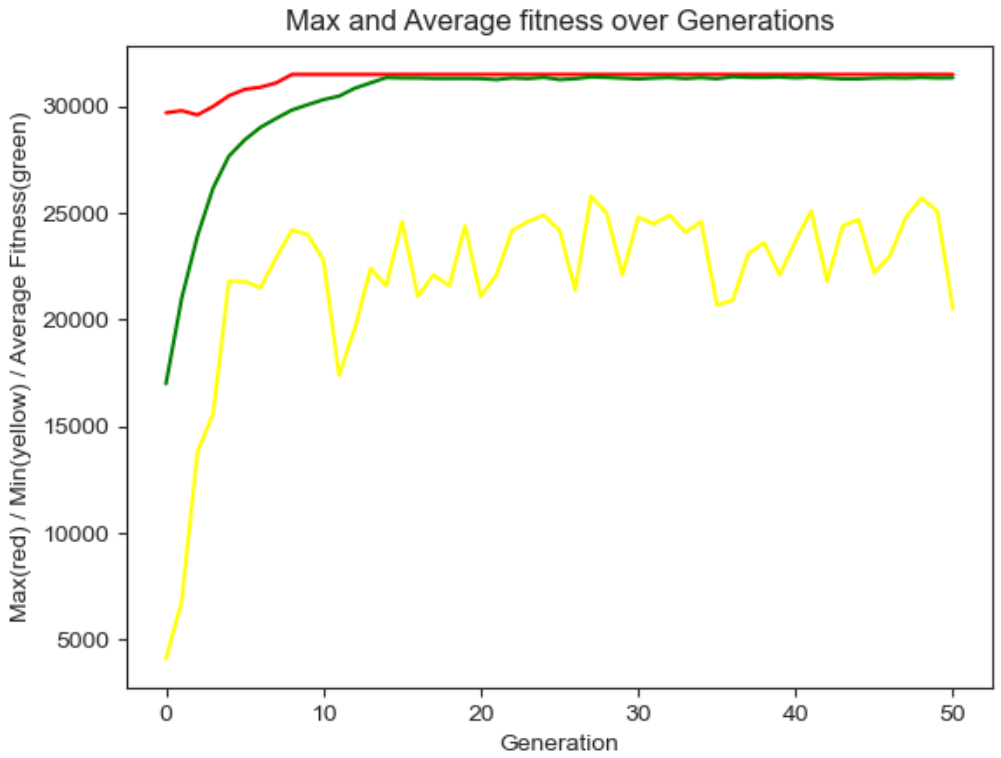
In the **first case**, we have the following list of items:

```
- Adding passport: weight = 100, value = 5000, accumulated weight
= 100, accumulated value = 5000
- Adding money: weight = 50, value = 5000, accumulated weight =
150, accumulated value = 10000
- Adding phone: weight = 300, value = 1000, accumulated weight =
450, accumulated value = 11000
- Adding laptop: weight = 1000, value = 5000, accumulated weight =
1450, accumulated value = 16000
- Adding laptop charger: weight = 300, value = 5000, accumulated
weight = 1750, accumulated value = 21000
- Adding mint: weight = 100, value = 500, accumulated weight =
1850, accumulated value = 21500
- Adding watch: weight = 100, value = 800, accumulated weight =
1950, accumulated value = 22300
- Adding headphones: weight = 200, value = 800, accumulated weight
= 2150, accumulated value = 23100
- Adding book: weight = 100, value = 700, accumulated weight =
2250, accumulated value = 23800
- Adding notebook: weight = 100, value = 1000, accumulated weight
= 2350, accumulated value = 24800
- Adding pen: weight = 50, value = 1000, accumulated weight =
2400, accumulated value = 25800
- Adding pencil: weight = 50, value = 1000, accumulated weight =
2450, accumulated value = 26800
- Adding eraser: weight = 50, value = 1000, accumulated weight =
2500, accumulated value = 27800
```

```
- Adding calculator: weight = 100, value = 700, accumulated weight
= 2600, accumulated value = 28500
- Adding pencil case: weight = 100, value = 700, accumulated
weight = 2700, accumulated value = 29200
- Adding lunchbox: weight = 100, value = 600, accumulated weight =
2800, accumulated value = 29800
- Adding sandwich: weight = 100, value = 700, accumulated weight =
2900, accumulated value = 30500
- Adding snacks: weight = 100, value = 1000, accumulated weight =
3000, accumulated value = 31500

- Skipping water bottle
- Skipping sunglasses
- Skipping first aid
- Skipping phone charger
- Total weight = 3000, Total value = 31500
```

The graph will be:

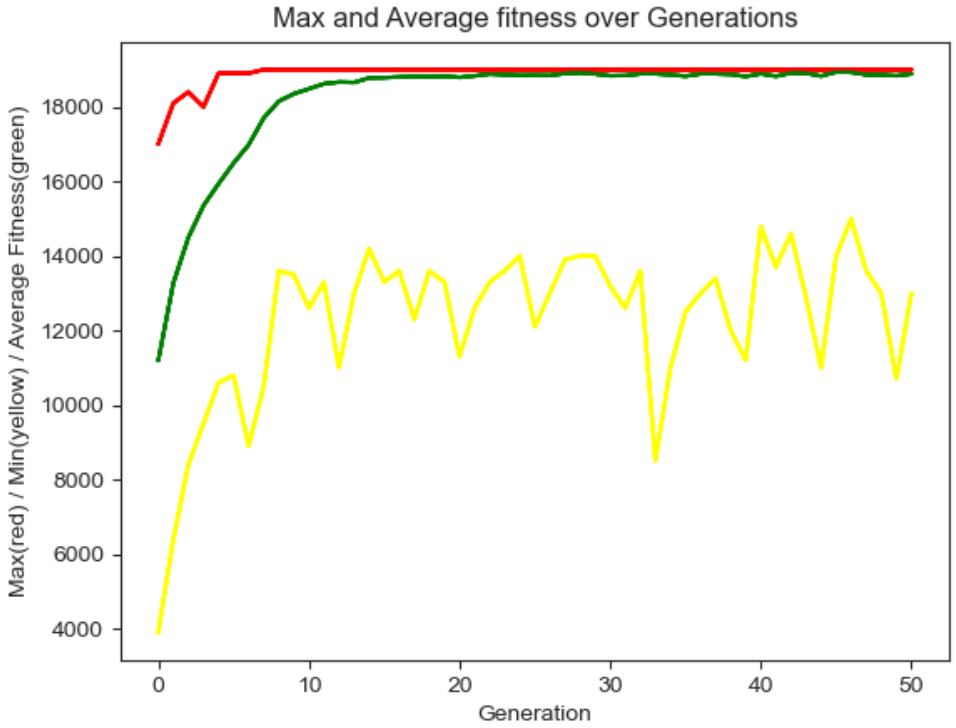


In the **second case**, we have the following list of items:

```
- Adding passport: weight = 100, value = 1000, accumulated weight
= 100, accumulated value = 1000
- Adding laptop: weight = 1000, value = 5000, accumulated weight =
1100, accumulated value = 6000
- Adding laptop charger: weight = 300, value = 5000, accumulated
weight = 1400, accumulated value = 11000
- Adding money: weight = 50, value = 1500, accumulated weight =
1450, accumulated value = 12500
- Adding watch: weight = 100, value = 800, accumulated weight =
1550, accumulated value = 13300
- Adding notebook: weight = 100, value = 1000, accumulated weight
= 1650, accumulated value = 14300
- Adding pen: weight = 50, value = 1000, accumulated weight =
1700, accumulated value = 15300
- Adding pencil: weight = 50, value = 1000, accumulated weight =
1750, accumulated value = 16300
- Adding eraser: weight = 50, value = 1000, accumulated weight =
1800, accumulated value = 17300
- Adding pencil case: weight = 100, value = 700, accumulated
weight = 1900, accumulated value = 18000
- Adding snacks: weight = 100, value = 1000, accumulated weight =
2000, accumulated value = 19000

- Skipping water bottle
- Skipping phone
- Skipping sunglasses
- Skipping first aid
- Skipping phone charger
- Skipping chapstick
- Skipping mint
- Skipping headphones
- Skipping book
- Skipping calculator
- Skipping lunch box
- Skipping sandwich
- Total weight = 2000, Total value = 19000
```

The graph will be:



In the **third case**, we have the following list of items:

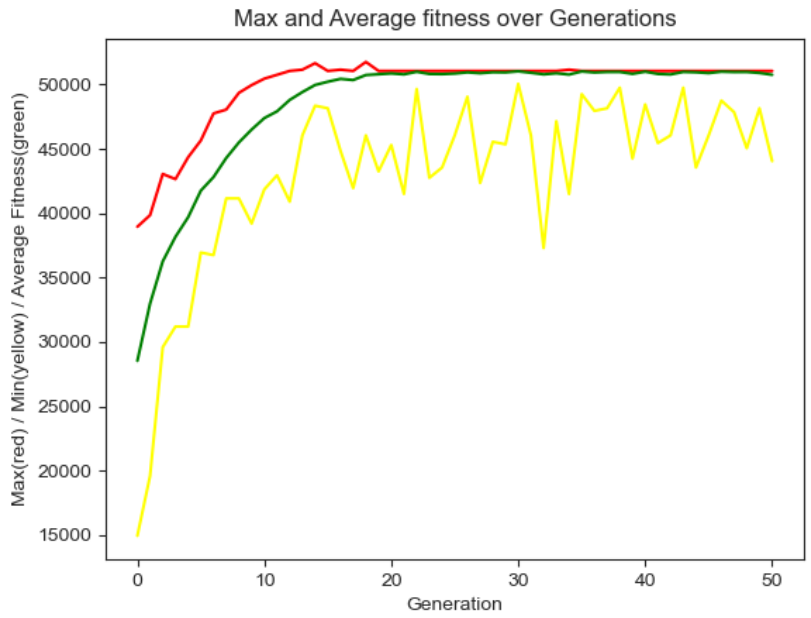
```
- Adding passport: weight = 100, value = 5000, accumulated weight
= 100, accumulated value = 5000
- Adding money: weight = 50, value = 5000, accumulated weight =
150, accumulated value = 10000
- Adding phone: weight = 300, value = 2000, accumulated weight =
450, accumulated value = 12000
- Adding laptop charger: weight = 300, value = 5000, accumulated
weight = 750, accumulated value = 17000
- Adding first aid: weight = 300, value = 2000, accumulated weight
= 1050, accumulated value = 19000
- Adding watch: weight = 100, value = 800, accumulated weight =
1150, accumulated value = 19800
- Adding lunch box: weight = 100, value = 600, accumulated weight
= 1250, accumulated value = 20400
- Adding sandwich: weight = 100, value = 700, accumulated weight =
1350, accumulated value = 21100
- Adding snacks: weight = 100, value = 1000, accumulated weight =
1450, accumulated value = 22100
- Adding underwear: weight = 200, value = 1300, accumulated weight
```

= 1650, accumulated value = 23400
- Adding slepping bag: weight = 1000, value = 2000, accumulated weight = 2650, accumulated value = 25400
- Adding camera charger: weight = 300, value = 800, accumulated weight = 2950, accumulated value = 26200
- Adding map: weight = 100, value = 1500, accumulated weight = 3050, accumulated value = 27700
- Adding toothbrush: weight = 100, value = 600, accumulated weight = 3150, accumulated value = 28300
- Adding toothpaste: weight = 100, value = 600, accumulated weight = 3250, accumulated value = 28900
- Adding shower gel: weight = 100, value = 600, accumulated weight = 3350, accumulated value = 29500
- Adding soap: weight = 100, value = 600, accumulated weight = 3450, accumulated value = 30100
- Adding hair gel: weight = 100, value = 600, accumulated weight = 3550, accumulated value = 30700
- Adding umbrella: weight = 100, value = 700, accumulated weight = 3650, accumulated value = 31400
- Adding headphones: weight = 100, value = 600, accumulated weight = 3750, accumulated value = 32000
- Adding book: weight = 100, value = 700, accumulated weight = 3850, accumulated value = 32700
- Adding pen: weight = 100, value = 500, accumulated weight = 3950, accumulated value = 33200
- Adding lunch box: weight = 100, value = 600, accumulated weight = 4050, accumulated value = 33800
- Adding compass: weight = 100, value = 900, accumulated weight = 4150, accumulated value = 34700
- Adding knife: weight = 100, value = 900, accumulated weight = 4250, accumulated value = 35600
- Adding torch: weight = 100, value = 900, accumulated weight = 4350, accumulated value = 36500
- Adding firewood: weight = 100, value = 400, accumulated weight = 4450, accumulated value = 36900
- Adding sandwich: weight = 100, value = 500, accumulated weight = 4550, accumulated value = 37400
- Adding emergency whistle : weight = 100, value = 600, accumulated weight = 4650, accumulated value = 38000
- Adding waterproof bag: weight = 100, value = 1000, accumulated weight = 4750, accumulated value = 39000
- Adding waterproof clothes: weight = 100, value = 1000,

accumulated weight = 4850, accumulated value = 40000
- Adding snacks: weight = 100, value = 1000, accumulated weight = 4950, accumulated value = 41000
- Adding travel pilows: weight = 100, value = 800, accumulated weight = 5050, accumulated value = 41800
- Adding guidebook: weight = 100, value = 750, accumulated weight = 5150, accumulated value = 42550
- Adding clothes to sleep: weight = 100, value = 1300, accumulated weight = 5250, accumulated value = 43850
- Adding waterproof shoes: weight = 100, value = 1300, accumulated weight = 5350, accumulated value = 45150
- Adding waterproof socks: weight = 100, value = 1300, accumulated weight = 5450, accumulated value = 46450
- Adding waterproof jacket: weight = 100, value = 1300, accumulated weight = 5550, accumulated value = 47750
- Adding waterproof pants: weight = 100, value = 1300, accumulated weight = 5650, accumulated value = 49050
- Adding waterproof shorts: weight = 100, value = 800, accumulated weight = 5750, accumulated value = 49850
- Adding waterproof hat: weight = 100, value = 900, accumulated weight = 5850, accumulated value = 50750
- Adding waterproof gloves: weight = 100, value = 1000, accumulated weight = 5950, accumulated value = 51750

- Skipping water bottle
- Skipping laptop
- Skipping sunglasses
- Skipping tablet
- Skipping camera
- Skipping phone charger
- Skipping towel
- Total weight = 5950, Total value = 51750

The graph will be:



In the **fourth case**, we have the following list of items:

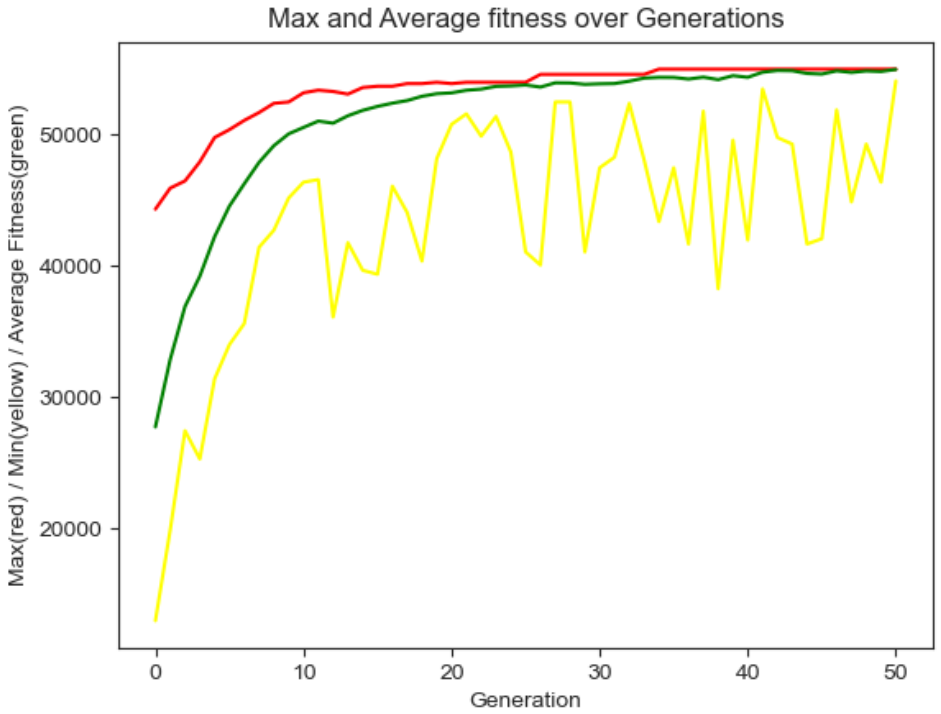
```
- Adding passport: weight = 100, value = 5000, accumulated weight
= 100, accumulated value = 5000
- Adding money: weight = 50, value = 5000, accumulated weight =
150, accumulated value = 10000
- Adding chapstick: weight = 100, value = 1000, accumulated weight
= 250, accumulated value = 11000
- Adding laptop charger: weight = 300, value = 5000, accumulated
weight = 550, accumulated value = 16000
- Adding first aid: weight = 300, value = 1500, accumulated weight
= 850, accumulated value = 17500
- Adding sandwich: weight = 100, value = 700, accumulated weight =
950, accumulated value = 18200
- Adding snacks: weight = 100, value = 1000, accumulated weight =
1050, accumulated value = 19200
- Adding map: weight = 100, value = 1500, accumulated weight =
1150, accumulated value = 20700
- Adding phone charger: weight = 300, value = 1500, accumulated
weight = 1450, accumulated value = 22200
- Adding toothbrush: weight = 100, value = 600, accumulated weight
```

= 1550, accumulated value = 22800
- Adding toothpaste: weight = 100, value = 600, accumulated weight = 1650, accumulated value = 23400
- Adding soap: weight = 100, value = 600, accumulated weight = 1750, accumulated value = 24000
- Adding headphones: weight = 100, value = 600, accumulated weight = 1850, accumulated value = 24600
- Adding book: weight = 100, value = 700, accumulated weight = 1950, accumulated value = 25300
- Adding compass: weight = 100, value = 900, accumulated weight = 2050, accumulated value = 26200
- Adding knife: weight = 100, value = 900, accumulated weight = 2150, accumulated value = 27100
- Adding torch: weight = 100, value = 900, accumulated weight = 2250, accumulated value = 28000
- Adding sandwich: weight = 100, value = 500, accumulated weight = 2350, accumulated value = 28500
- Adding waterproof bag: weight = 100, value = 1000, accumulated weight = 2450, accumulated value = 29500
- Adding waterproof clothes: weight = 100, value = 1000, accumulated weight = 2550, accumulated value = 30500
- Adding snacks: weight = 100, value = 1000, accumulated weight = 2650, accumulated value = 31500
- Adding travel pilows: weight = 100, value = 800, accumulated weight = 2750, accumulated value = 32300
- Adding guidebook: weight = 100, value = 750, accumulated weight = 2850, accumulated value = 33050
- Adding clothes to sleep: weight = 100, value = 1300, accumulated weight = 2950, accumulated value = 34350
- Adding waterproof shoes: weight = 100, value = 1300, accumulated weight = 3050, accumulated value = 35650
- Adding waterproof socks: weight = 100, value = 1300, accumulated weight = 3150, accumulated value = 36950
- Adding waterproof jacket: weight = 100, value = 1300, accumulated weight = 3250, accumulated value = 38250
- Adding waterproof pants: weight = 100, value = 1300, accumulated weight = 3350, accumulated value = 39550
- Adding waterproof hat: weight = 100, value = 900, accumulated weight = 3450, accumulated value = 40450
- Adding waterproof glove: weight = 100, value = 1000, accumulated weight = 3550, accumulated value = 41450
- Adding hair brush: weight = 100, value = 1500, accumulated

weight = 3650, accumulated value = 42950
- Adding hair dryer: weight = 100, value = 1500, accumulated weight = 3750, accumulated value = 44450
- Adding hair straightener: weight = 100, value = 1500, accumulated weight = 3850, accumulated value = 45950
- Adding hair curler: weight = 100, value = 1500, accumulated weight = 3950, accumulated value = 47450
- Adding hair spray: weight = 100, value = 1500, accumulated weight = 4050, accumulated value = 48950
- Adding hair gel: weight = 100, value = 1500, accumulated weight = 4150, accumulated value = 50450
- Adding hair wax: weight = 100, value = 1500, accumulated weight = 4250, accumulated value = 51950
- Adding hair mousse: weight = 100, value = 1500, accumulated weight = 4350, accumulated value = 53450
- Adding hair conditioner: weight = 100, value = 1500, accumulated weight = 4450, accumulated value = 54950

- Skipping water bottle
- Skipping phone
- Skipping laptop
- Skipping sunglasses
- Skipping watch
- Skipping lunch box
- Skipping tablet
- Skipping underwear
- Skipping camera
- Skipping slepping bag
- Skipping camera charger
- Skipping shower gel
- Skipping towel
- Skipping hair gel
- Skipping umbrella
- Skipping pen
- Skipping lunch box
- Skipping firewood
- Skipping emergency whistle
- Skipping waterproof shorts
- Total weight = 4450, Total value = 54950

The graph will be:



In the **fourth case** if I set the *population* and *generation* per 1000. We can see this result:

```
- Adding passport: weight = 100, value = 5000, accumulated weight
= 100, accumulated value = 5000
- Adding money: weight = 50, value = 5000, accumulated weight =
150, accumulated value = 10000
- Adding chapstick: weight = 100, value = 1000, accumulated weight
= 250, accumulated value = 11000
- Adding laptop charger: weight = 300, value = 5000, accumulated
weight = 550, accumulated value = 16000
- Adding watch: weight = 100, value = 800, accumulated weight =
650, accumulated value = 16800
- Adding lunch box: weight = 100, value = 600, accumulated weight
= 750, accumulated value = 17400
- Adding sandwich: weight = 100, value = 700, accumulated weight =
850, accumulated value = 18100
- Adding snacks: weight = 100, value = 1000, accumulated weight =
950, accumulated value = 19100
```

- Adding underwear: weight = 200, value = 1300, accumulated weight = 1150, accumulated value = 20400
- Adding map: weight = 100, value = 1500, accumulated weight = 1250, accumulated value = 21900
- Adding toothbrush: weight = 100, value = 600, accumulated weight = 1350, accumulated value = 22500
- Adding toothpaste: weight = 100, value = 600, accumulated weight = 1450, accumulated value = 23100
- Adding towel: weight = 100, value = 600, accumulated weight = 1550, accumulated value = 23700
- Adding umbrella: weight = 100, value = 700, accumulated weight = 1650, accumulated value = 24400
- Adding book: weight = 100, value = 700, accumulated weight = 1750, accumulated value = 25100
- Adding lunch box: weight = 100, value = 600, accumulated weight = 1850, accumulated value = 25700
- Adding compass: weight = 100, value = 900, accumulated weight = 1950, accumulated value = 26600
- Adding knife: weight = 100, value = 900, accumulated weight = 2050, accumulated value = 27500
- Adding torch: weight = 100, value = 900, accumulated weight = 2150, accumulated value = 28400
- Adding emergency whistle: weight = 100, value = 600, accumulated weight = 2250, accumulated value = 29000
- Adding waterproof bag: weight = 100, value = 1000, accumulated weight = 2350, accumulated value = 30000
- Adding waterproof clothes: weight = 100, value = 1000, accumulated weight = 2450, accumulated value = 31000
- Adding snacks: weight = 100, value = 1000, accumulated weight = 2550, accumulated value = 32000
- Adding travel pilows: weight = 100, value = 800, accumulated weight = 2650, accumulated value = 32800
- Adding guidebook: weight = 100, value = 750, accumulated weight = 2750, accumulated value = 33550
- Adding clothes to sleep: weight = 100, value = 1300, accumulated weight = 2850, accumulated value = 34850
- Adding waterproof shoes: weight = 100, value = 1300, accumulated weight = 2950, accumulated value = 36150
- Adding waterproof socks: weight = 100, value = 1300, accumulated weight = 3050, accumulated value = 37450
- Adding waterproof pants: weight = 100, value = 1300, accumulated weight = 3250, accumulated value = 40050

- Adding waterproof short: weight = 100, value = 800, accumulated weight = 3350, accumulated value = 40850
- Adding waterproof hat: weight = 100, value = 900, accumulated weight = 3450, accumulated value = 41750
- Adding waterproof gloves: weight = 100, value = 1000, accumulated weight = 3550, accumulated value = 42750
- Adding hair brush: weight = 100, value = 1500, accumulated weight = 3650, accumulated value = 44250
- Adding hair dryer: weight = 100, value = 1500, accumulated weight = 3750, accumulated value = 45750
- Adding hair straightener: weight = 100, value = 1500, accumulated weight = 3850, accumulated value = 47250
- Adding hair curler: weight = 100, value = 1500, accumulated weight = 3950, accumulated value = 48750
- Adding hair spray: weight = 100, value = 1500, accumulated weight = 4050, accumulated value = 50250
- Adding hair gel: weight = 100, value = 1500, accumulated weight = 4150, accumulated value = 51750
- Adding hair wax: weight = 100, value = 1500, accumulated weight = 4250, accumulated value = 53250
- Adding hair mousse: weight = 100, value = 1500, accumulated weight = 4350, accumulated value = 54750
- Adding hair conditioner: weight = 100, value = 1500, accumulated weight = 4450, accumulated value = 56250

- Skipping water bottle
- Skipping phone
- Skipping laptop
- Skipping sunglasses
- Skipping first aid
- Skipping tablet
- Skipping camera
- Skipping slepping bag
- Skipping camera charger
- Skipping phone charger
- Skipping shower gel
- Skipping soap
- Skipping hair gel
- Skipping headphones
- Skipping pen
- Skipping firewood
- Skipping sandwich

```
- Total weight = 4450, Total value = 56250
```

The graph will be:

In the second version, the following items were added:

- Underwear
- Waterproof shorts
- Emergency whistle
- Lunch box
- Umbrella
- Towel
- Watch

And the result was 56250 against 54950.

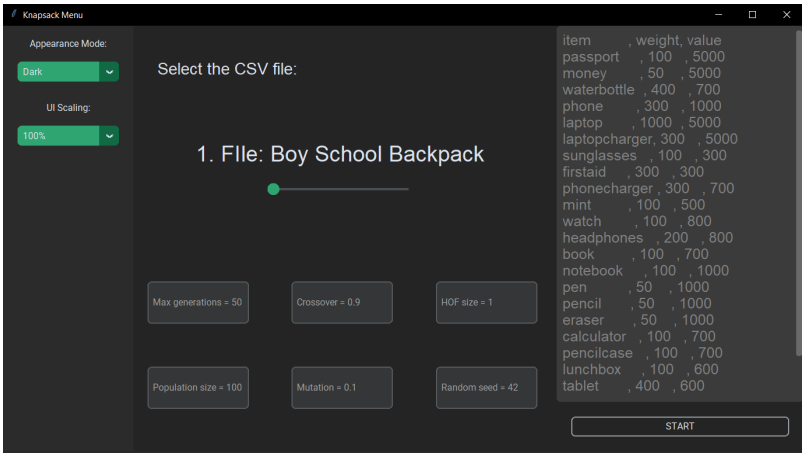
6. Console application

We used Custom Tkinter to create this application:

Tkinter is a Python Package for creating GUI applications.
Python has a lot of GUI frameworks, but Tkinter is the only framework that's built into the Python standard library.

Tkinter has several strengths:

- it's cross-platform, so the same code works on Windows, macOS, and Linux.
- Tkinter is lightweight and relatively painless to use compared to other frameworks.



This makes it a compelling choice for building GUI applications in Python, especially for applications where a modern shine is unnecessary, and the top priority is to build something functional and cross-platform quickly.

An output of this program shows us information about:

- Best individual items
- Best fitness value
- Items added to backpack
- Items that were skipped
- Total weight and the total value of the problem

And also creates a plot with max, average, and min values of the fitness:

```
-- Best Ever Individual =  [1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

-- Best Ever Fitness =  31500.0

-- Knapsack Items:

- Adding passport     : weight = 100, value = 5000, accumulated weight = 100, accumulated value = 5000
- Adding money        : weight = 50, value = 5000, accumulated weight = 150, accumulated value = 10000
- Adding phone        : weight = 300, value = 1000, accumulated weight = 450, accumulated value = 11000
- Adding laptop       : weight = 1000, value = 5000, accumulated weight = 1450, accumulated value = 16000
- Adding laptopcharger: weight = 300, value = 5000, accumulated weight = 1750, accumulated value = 21000
- Adding mint         : weight = 100, value = 500, accumulated weight = 1850, accumulated value = 21500
- Adding watch        : weight = 100, value = 800, accumulated weight = 1950, accumulated value = 22300
- Adding headphones   : weight = 200, value = 800, accumulated weight = 2150, accumulated value = 23100
- Adding book         : weight = 100, value = 700, accumulated weight = 2250, accumulated value = 23800
- Adding notebook     : weight = 100, value = 1000, accumulated weight = 2350, accumulated value = 24800
- Adding pen          : weight = 50, value = 1000, accumulated weight = 2400, accumulated value = 25800
- Adding pencil       : weight = 50, value = 1000, accumulated weight = 2450, accumulated value = 26800
- Adding eraser       : weight = 50, value = 1000, accumulated weight = 2500, accumulated value = 27800
- Adding calculator   : weight = 100, value = 700, accumulated weight = 2600, accumulated value = 28500
- Adding pencilcase   : weight = 100, value = 700, accumulated weight = 2700, accumulated value = 29200
- Adding lunchbox     : weight = 100, value = 600, accumulated weight = 2800, accumulated value = 29800
- Adding sandwich     : weight = 100, value = 700, accumulated weight = 2900, accumulated value = 30500
- Adding snacks       : weight = 100, value = 1000, accumulated weight = 3000, accumulated value = 31500
Done!
- Skipping waterbottle
- Skipping sunglasses
- Skipping firstaid
- Skipping phonecharger
- Total weight = 3000, Total value = 31500
```