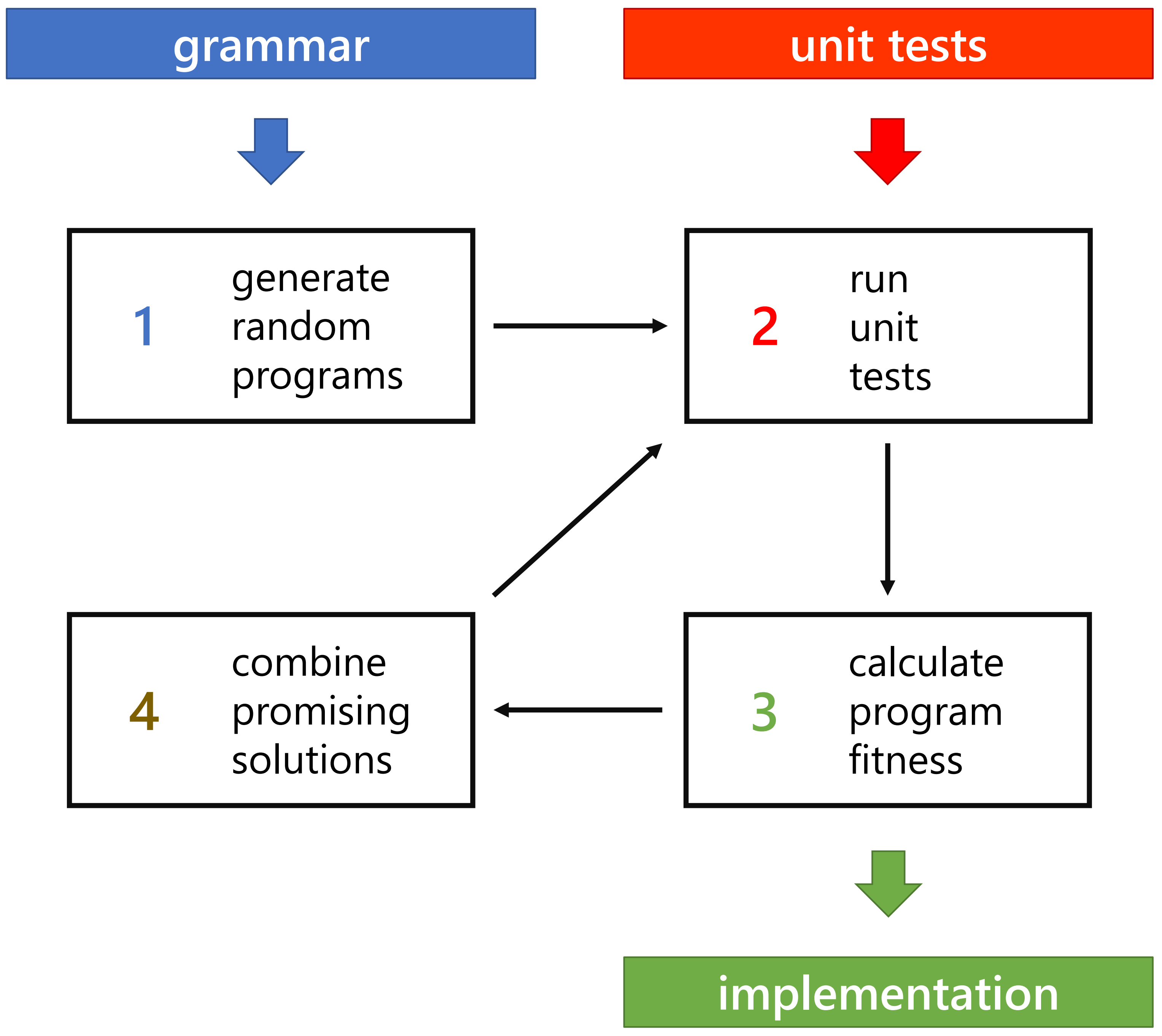
Excel@FIT 2017 Jan Svoboda 3BIT **62**

Automating Test Driven Development with Grammatical Evolution

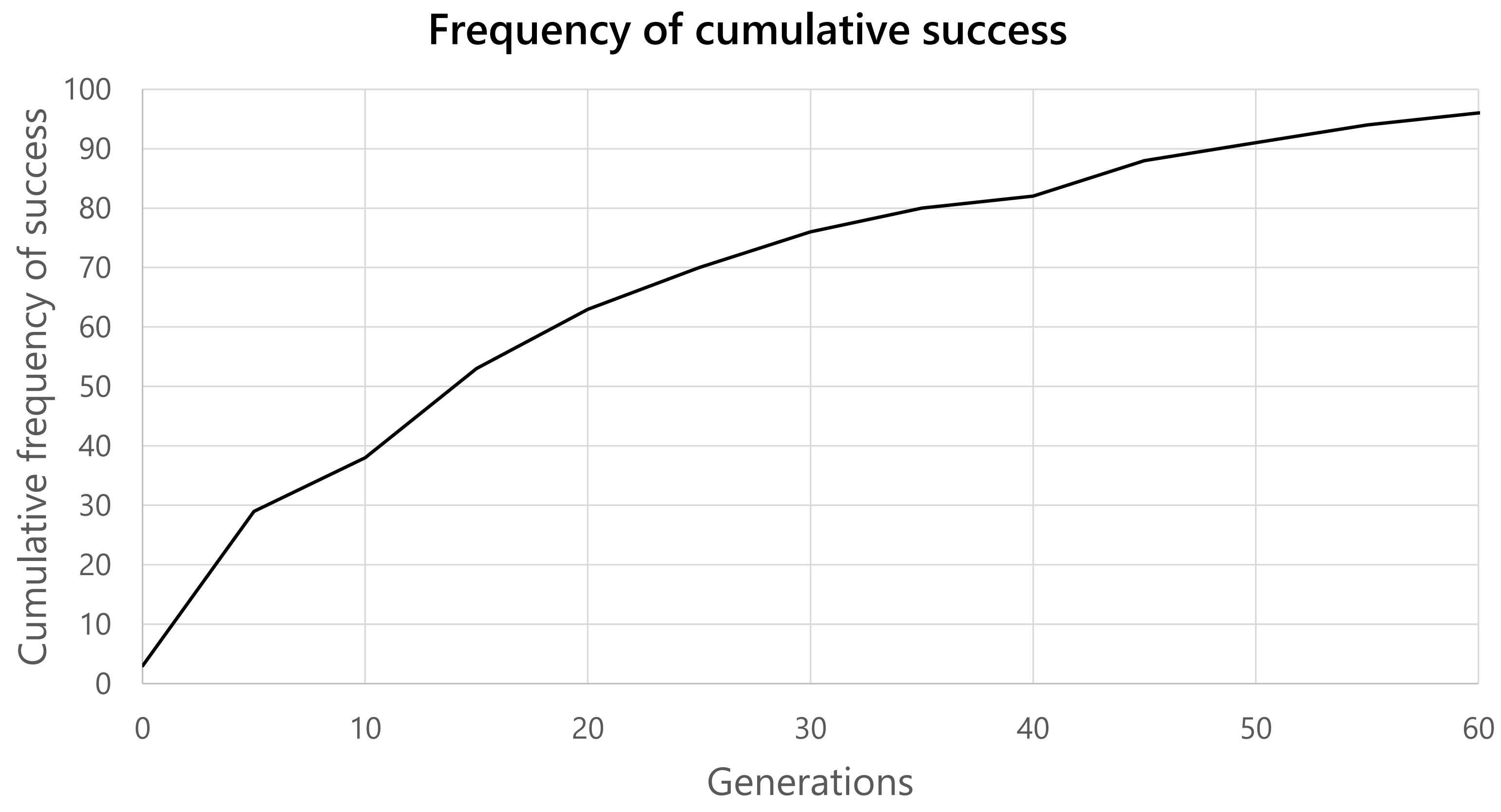
Summary

The goal of this work is to introduce a system able to generate implementation of functions in a programming language based solely on unit tests. This is achieved by creating a new grammatical evolution library called **Gram** and integrating it with the testing ecosystem of the PHP language.

System overview


grammar
unit tests
1 generate random programs
2 run unit tests
3 calculate program fitness
4 combine promising solutions
implementation

Experiment: results



```
<?php

function array_filter($input, $filter) {
    $output = [];

    foreach ($input as $item) {
        if ($filter($item)) {
            $output[] = $item;
        }
    }

    return $output;
}
```

Fitness calculation

Fitness of candidate solutions in a single test assertion is calculated using the following equations:

Numeric types $dist(e,a) = |e - a|$

Boolean types $dist(e,a) = \begin{cases} 0, & if\ e = a. \\ C, & otherwise. \end{cases}$

String types $dist(e,a) = lev(e,a)$

Array types $dist(e,a) = \sum dist(e_i, a_i)$

where e is the value expected by the test and a is the actual value the program returned.

Experiment: parameters

Goal: create function **`array_filter`**`($input, $filter)`

Tests

```
$filter = function ($item) { return $item > 0;}
```

Input	Correct output
`[]`	`[]`
`[-10, -5, -3, -1]`	`[]`
`[-10, -1, 3, 5]`	`[3, 5]`
`[1, 20, 42]`	`[1, 20, 42]`

Evolution parameters

Parameter	**Value**
Initialization	Random
Genotype length	Fixed, 40 codons
Population size	200 individuals
Selection	Tournament, size: 5
Crossover	One-point, probability: 1.0
Mutation	Codon-level, probability: 0.15
Success predicate	Fitness is equal to 0

```
<?php

function array_filter($input, $filter) {
    $output = [];           // dead code
    $output = [];

    foreach ($input as $item) {
        if ($filter($item)) {
            $output[] = $item;
        }
    }

    return $output;

    foreach ($input as $item) {
        return $output;
    }
}
```

Supervisor: prof. Lukáš Sekanina