

# Evolučný algoritmus

## Zenova záhrada

### Zadanie:

Uvedenú úlohu riešite pomocou evolučného algoritmu. Maximálny počet génov nesmie presiahnuť polovicu obvodu záhrady plus počet kameňov, v našom prípade podľa prvého obrázku  $12+10+6=28$ . Fitnes je určená počtom pohrabaných políčok. Výstupom je matica, znázorňujúca cesty mnícha. Je potrebné, aby program zvládol aspoň záhradku podľa prvého obrázku, ale vstupom môže byť v princípe ľubovoľná mapa.

### Riešenie:

Zo zadania sme vieme maximálnu dĺžku chomozómu, naplníme ho súradnicami ktoré sú reprezentované na príklade zo zadania na obrázku.

	1	2	3	4	5	6	7	8	9	10	11	12	
35													13
36													14
37													15
38													16
39													17
40													18
41													19
42													20
43													21
44													22
	23	24	25	26	27	28	29	30	31	32	33	34	

### Opis algoritmu

Pred spustením programu, si v triede GenetickéHľadanie nastavíme hodnotu konštánt ktoré sa nebudú meniť počas vykonávania kódu. Sú to:

- VELKOSŤ\_POPULÁCIE
- MAX\_GENERÁCI
- METÓDA\_VÝBERU
- METÓDA\_KRÍŽENIA
- MIN\_MUTÁCIA
- MAX\_MUTÁCIA

Tieto premenné nám budú meniť ako sa bude program správať počas hľadania riešenia na zadaný problém. Výber problému môžeme meniť v triede Zaciatok, kde zmeníme názov vstupného súboru.

Ako prvé čo program spravý prečíta vstup zo súboru v ktorom je výška a šírka záhrady, a po riadkoch sú napísane súradnice jednotlivých kameňov. Takto vytvoríme záhradu. Túto záhradu pošleme triede GenetickeHľadanie, ktorá začne evolučný algoritmus.

Ako prvé sa vytvorí populácia zadanej veľkosti, každému jedincovi sa náhodne nakonfiguruje chromozóm. Potom každého jedinca posielame aby pohrabal záhradu.

Pohrabanie záhrady prebieha tak, že zoberieme prvé číslo v chromozóme a pozrieme sa na to políčko či tam môžeme vojsť. Ak môžeme tak hrabeme rovno až po koniec záhrady. Ak narazíme na prekážku(kameň, alebo už pohrabanú časť), tak sa s pravdepodobnosťou 50% jedinec rozhodne kde sa má otočiť. Ak aj do toho smeru narazí naprekážku tak vyberie opačný smer. Ak aj tam narazí na prekážku tak skončí s hrabaním a vráti koľko políčok pohrabal. Toto je jeho fitness. Ak počas hrabania narazíme na jedinca ktorý pohrabal celú záhradu tak vypíšeme jeho záhradu. Problém pri mojom algoritme je, že ak toho istého jedinca znovu pošleme na pohrabanie záhrady tak ju nemusí pohrabat'. Lebo kde sa otočí určuje náhoda 50%.

Ak všetci jedinci v populácii pohrabali záhradu, tak si zistíme maximálnu a minimálnu fitness hodnotu každého. Ak sme našli jedinca ktorý pohrabal celú záhradu, tak sa program skončí. Ak taký prípad nenájde tak ďalej pokračuje vo vytváraní novej generácie.

Vytvorenie novej generácie prebieha tak, že najprv každému jedincovi znížime hodnotu fitness o najmenšiu fitness v generácii. Tým zaistíme že sa nám nebudú krížiť jedinci ktorý nemali dobrý fitness.

Potom vyberáme jedincov podľa metódy výberu ktorú sme si vybrali na začiatku, a takisto metódu kríženia. Nižšie sú opísané podrobnejšie.

Počas kríženia sa môže stať mutácia. Mutácia je zmena jedného prvku v chromozóme, predtým ale zistíme či sa už taký prvok nachádza v chromozóme, ak áno tak ho iba vymeníme. Pravdepodobnosť mutácie s každým smutovaným génom stúpa, ale ak dosiahne stanovenú „strechu“ tak sa vráti na začiatočnú hodnotu.

Na konci kríženia ešte nastáva elitizmus kde vyberieme najlepšieho člena predchádzajúcej generácie a vložíme ho do novej.

## Metódy výberu

### Ruleta

Ruleta spočíva vo výbere náhodnej percentuálnej hodnoty zo sumy fitness. Potom prechádzame hodnoty fitness až pokým nenájdeme hodnotu ktorá je väčšia ako tá ktorú sme si vygenerovali. Vrátime index danej hodnoty.

## Turnaj

Táto metóda funguje tak že vyberieme náhodných dvoch jedincov z populácie a vyberieme z nich lepšieho.

## Metódy kríženia

### One-point

Metóda kríženia ktorá náhodne vyberie jeden bod v chromozóme, ktorý rozdelí dva chromozómy na dve časti. Tieto časti spojíme a vytvoríme dve nové deti.

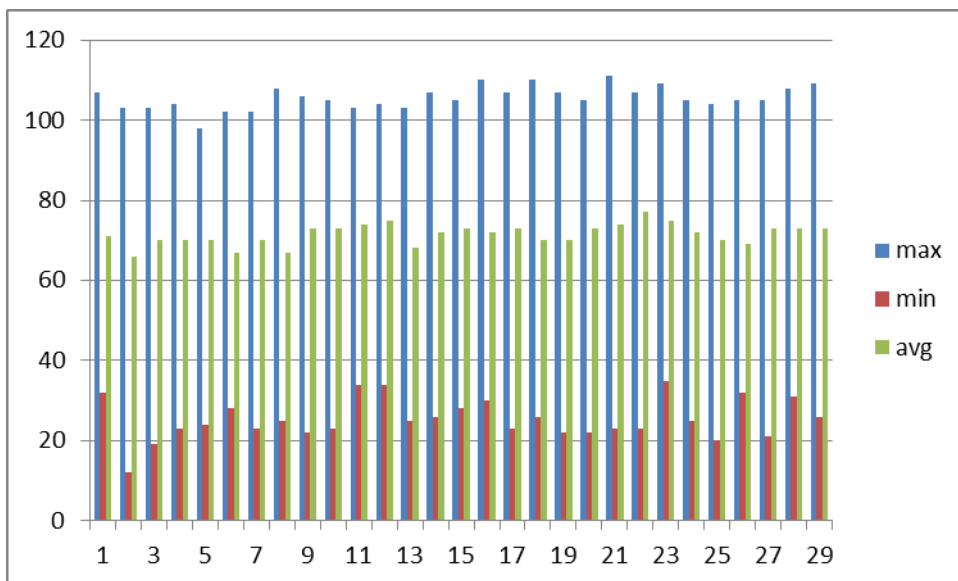
### Two-point

Táto metóda vyberie až dva body v ktoré rozdelia dva chromozómy na tri časti, strednú časť vymeníme a dostaneme

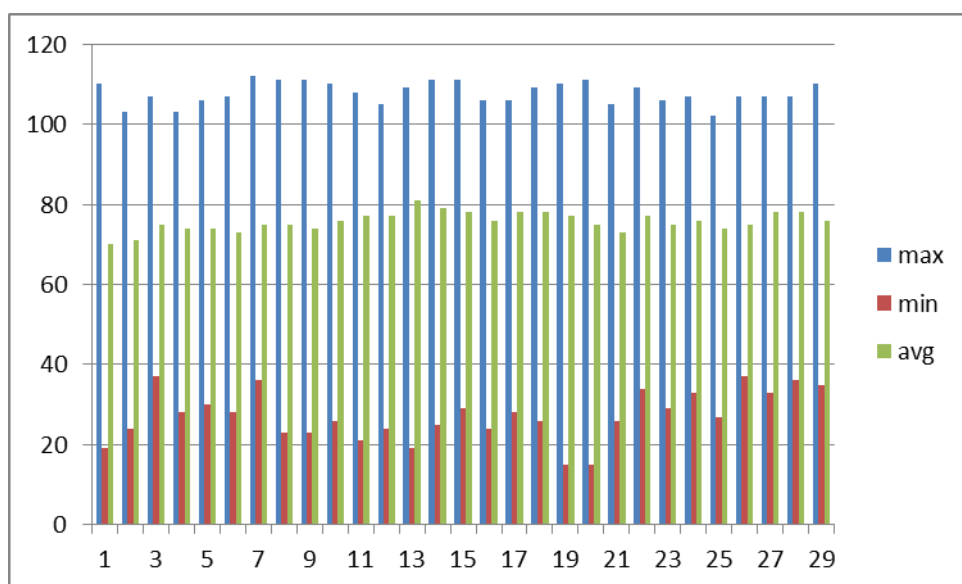
## Porovnanie metód

V týchto porovnaníach som dal maximálny počet generácií 29, a porovnával som ako sa program správa

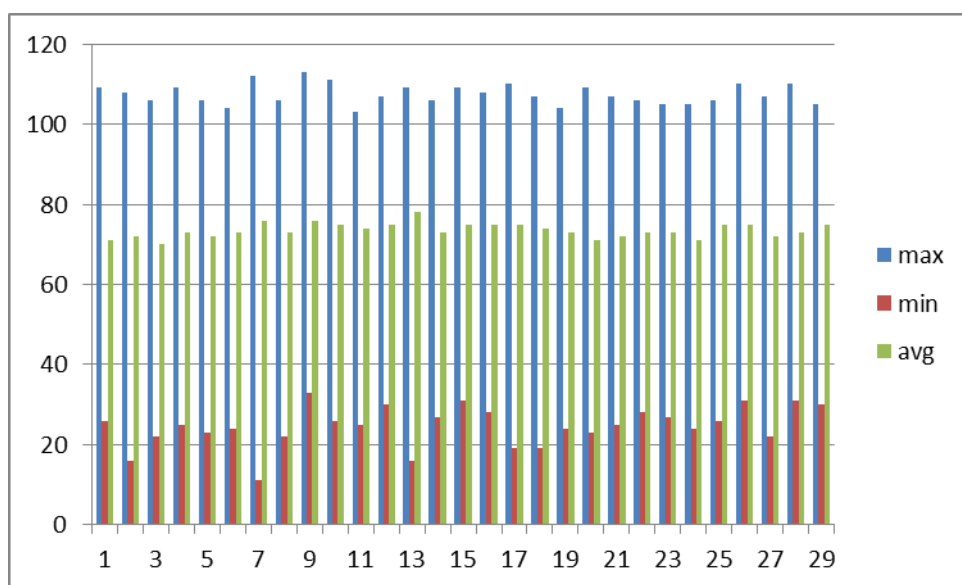
Rulete a one-point:



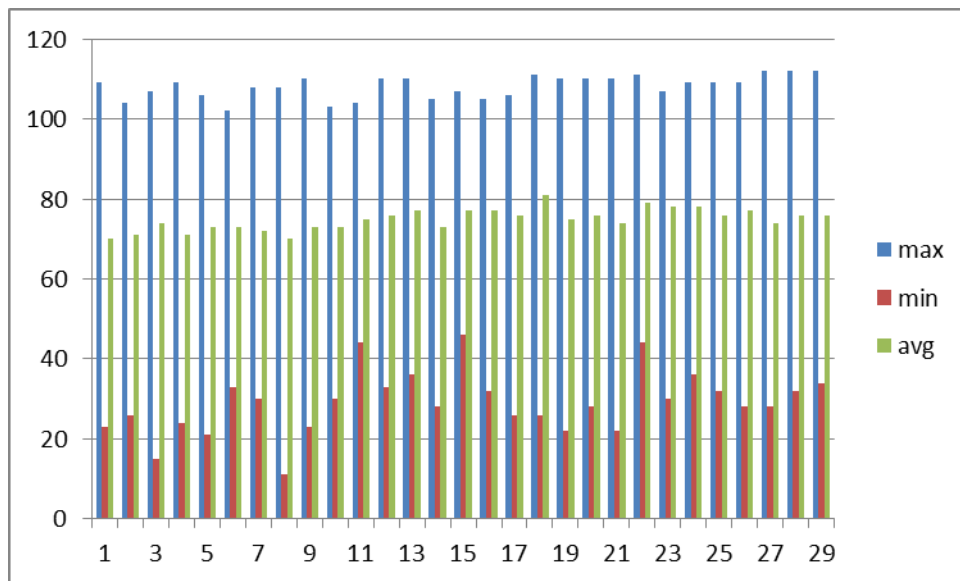
### Turnaj a one-point



### Ruleta a two-point



### Turnaj a two point



## Vylepšenie riešenia

- Vytvoriť viacero metód na križenie a výber
- Lepšia reprezentácia záhrady a jej prehľadovanie, žiadna náhoda