

**Genetické programování v
platformově nezávislém
programovacím jazyce**

**Genetic programming in platform
free programming language**

Tuto stránku nahradíte v tištěné verzi práce oficiálním zadáním Vaší diplomové či bakalářské práce.

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

Zde vložte text dohodnutého omezení přístupu k Vaší práci, chránící například firemní know-how. Zde vložte text dohodnutého omezení přístupu k Vaší práci, chránící například firemní know-how. A zavazujete se, že

1. o práci nikomu neřeknete,
2. po obhajobě na ni zapomenete a
3. budete popírat její existenci.

A ještě jeden důležitý odstavec. A ještě jeden důležitý odstavec. A ještě jeden důležitý odstavec. A ještě jeden důležitý odstavec. A ještě jeden důležitý odstavec. Konec textu dohodnutého omezení přístupu k Vaší práci.

V Ostravě 16. dubna 2009

+++
.....

Prohlašuji, že jsem tuto bakalářskou práci vypracovala samostatně. Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.

V Ostravě 16. dubna 2009

+++
.....

Rád bych na tomto místě poděkovala všem, kteří mi s prací pomohli, protože bez nich by tato práce nevznikla.

Abstrakt

Genetické programování a vůbec, genetické algoritmy jako pojem vznikly už před x lety a sklízí velké oblibě ve všech oblastech, která staví na inženýrských postupech.

Klíčová slova: genetický algoritmus, genom, populace, stochastický, evoluce

Abstract

This is English abstract. This is English abstract. This is English abstract. This is English abstract. This is English abstract. This is English abstract.

Keywords: typography, L^AT_EX, master thesis

Seznam použitých zkratek a symbolů

DVD	– Digital Versatile Disc
TNT	– Trinitrotoluen
OASIS	– Organization For The Advancement Of Structured Information Systems
HTML	– Hyper Text Markup Language

Obsah

1	Úvod	5
2	Evoluční algoritmy	6
2.1	Přehled algoritmů	6
3	Implementace frameworku	7
4	Ukázky sazby	8
4.1	Ukázka nadpisů	8
4.2	Sazba definic, vět atd.	8
4.3	Výpisy programů	9
4.4	Obrázky a tabulky	10
5	Závěr	14
6	Reference	15
	Přílohy	15
A	Grafy a měření	16

Seznam tabulek

1	Pokusná tabulka	12
2	Experimental Files — Detailed Statistics	13

Seznam obrázků

1	Pokusný obrázek – absolutní velikost	10
2	Pokusný obrázek – relativní velikost	10
3	Pokusný obrázek – otočený naležato	11
4	Nějaký graf	17

Seznam výpisů zdrojového kódu

1	Program v jazyce Java	9
2	Program v jazyce Java, načtený z externího souboru	9
3	Program v Pascalu	10

1 Úvod

Tento text je ukázkou sazby diplomové práce v \LaTeX u pomocí třídy dokumentů `diploma`. Pochopitelně text není skutečnou diplomovou prací, ale jen ukázkou použití implementovaných maker v praxi. V kapitole 4 jsou ukázky použití různých maker a prostředí. V kapitole 5 bude „jako závěr“. Zároveň tato kapitola slouží jako ukázka generování křížových odkazů v \LaTeX u.

2 Evoluční algoritmy

2.1 Přehled algoritmů

V této podkapitole si představíme seznam nejzákladnějších algoritmů, které patří do množiny evolučních algoritmů a našly si řadu využití v inženýrské praxi i ve světě jako takovém.

2.1.1 Genetický algoritmus

Genetický algoritmus GA je historicky prvním algoritmem, který byl kdy navržen.

3 Implementace frameworku

V této kapitole je nastíněna struktura frameworku s evolučními algoritmy, možnost použití a rozšíření v praxi.

3.1 Mutace

Mutace je jeden z evolučních operátorů, použitý v řadě algoritmů jako je např. Genetický algoritmus, Diferenciální evoluce, aj.

Reprezentace operátoru mutace je v knihovně JEvolutionCore v podobě generického rozhraní `MutateFunction` v balíčku `org.evolution.function`. Třídy implementující toto rozhraní musí definovat obsah metody `mutate`, která náhodně vybere řešení s pravděpodobností p a malinko změni hodnotu (hodnoty) v řešení. Knihovna má předdefinované 3 základní druhy mutací:

- Standardní mutace - mutace s pravděpodobností p pro výběr řešení, u kterého dojde ke změně jedné náhodně vybrané hodnoty genomu
- Vícebodová mutace - mutace s pravděpodobností p pro výběr řešení, u kterého dojde ke změně 1-n náhodně vybraných hodnot genomu
- Nahodilá mutace - mutace s pravděpodobností p pro náhodnou změnu napříč všemi hodnotami vybraných řešení

Standardní mutace je definovaná ve třídě `StandardMutateFunction` a její použití je především u Genetického algoritmu v základní podobě. Obecně je vhodná u řešení, s malou dimenzí, protože změna náhodně vybrané jedné hodnoty ještě udrží různorodost populace a algoritmus tak nestagneje.

4 Ukázky sazby

4.1 Ukázka nadpisů

Toto je nadpis podsekce, generováno makrem `\subsection`.

4.1.1 subsection

4.1.1.1 paragraph

4.1.1.1.1 subparagraph Ale tak hluboko se asi stejně nikdo nedostane.

4.2 Sazba definic, vět atd.

Určitě se bude hodit prostředí pro sazbu definice jako je definice binárního vyhledávacího stromu, viz definice 4.1.

Poznámka 4.1 Následující definice a věty nedávají dohromady příliš smysl. Jsou tu jen pro ukázkou.

Definice 4.1 *Binární strom je struktura definovaná nad konečnou množinou uzlů, která:*

- *neobsahuje žádný uzel,*
- *je složena ze tří disjunktních množin uzlů: kořene, binárního stromu zvaného levý podstrom a binárního stromu tzv. pravého podstromu.*

Pak by se taky mohla hodit nějaká věta a k ní důkaz.

Věta 4.1 *Průměrná časová složitost neúspěšného vyhledání v hashovací tabulce se separátním zřetězením je $\Theta(1 + \alpha)$, za předpokladu jednoduchého uniformního hashování.*

Důkaz. Za předpokladu jednoduchého uniformního hashování se každý klíč k hashuje se stejnou pravděpodobností do libovolného z m slotů tabulky. Průměrný čas neúspěšného hledání klíče k je proto průměrný čas prohledání jednoho z m seznamů. Průměrná délka každého takového seznamu je rovna faktorů naplnění $\alpha = n/m$. Tudíž lze očekávat, že budeme nuceni prozkoumat α prvků. Z toho plyne, že celkový čas pro neúspěšné hledání (plus navíc konstantní čas pro výpočet $h(k)$) je $\Theta(1 + \alpha)$. ■

Příklad 4.1

Mějme napsat funkci, která spočítá uzly ve stromu. Předpokládejme, že binární strom je definován způsobem uvedeným v definici 4.1 na straně 8. Naše úloha se výrazně zjednoduší uvědomíme-li si její rekurzivní charakter a předpokládáme, že aktuální uzel je R .

- Je-li R prázdný strom (tj. $R = NULL$), pak počet jeho uzlů je pochopitelně nula. Tím máme problém vyřešen.

- V opačném případě víme, že ve stromu určitě jeden uzel existuje (R) a počty uzlů v levém a pravém podstromu se dají určit obdobným způsobem rekurzivně. To znamená, že počet uzlů ve stromu s kořenem R je $1 + \text{pocet_uzlu}(A) + \text{pocet_uzlu}(B)$

Počty uzlů pro jednotlivé podstromy se předávají jako výsledky volání funkcí prostřednictvím zásobníku programu, nejsou tudíž potřeba žádné pomocné proměnné. ■

Poznámka 4.2 Program z příkladu 4.1 pochopitelně chybí, ale můžete se podívat třeba na program uvedený ve výpisu 1.

4.3 Výpisy programů

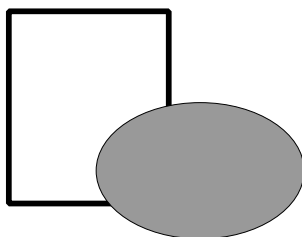
Tato diplomová práce má nastaven výchozí jazyk Java, jak je vidět z výpisu 1. Výpis kódu 1 zároveň demonstruje možnost přímého vložení zdrojového kódu programu do textu práce. Druhou možností je načtení zdrojového kódu programu z externího souboru, viz výpis 2. Pokud potřebujeme změnit programovací jazyk pro konkrétní výpis kódu, můžeme jeho to provést přímo v záhlaví prostředí `lstlisting`. Výpis 3 je v jazyku Pascal. Všimněte si zvýraznění klíčových slov.

Poznámka 4.3 Pro správnou sazbu je třeba pro odsazování používat tabulátory, nikoliv mezery.

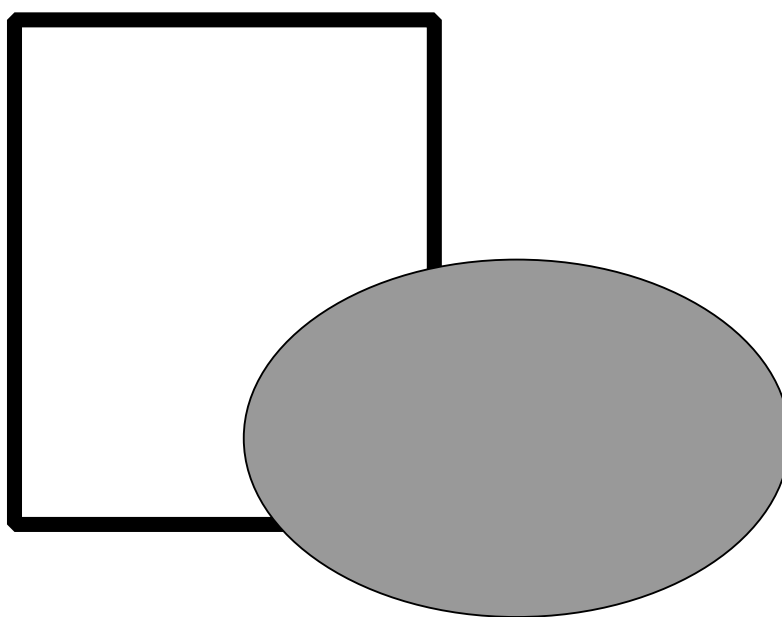
```
public class MyClass
{
    public int MyMethod(int a, int b)
    {
        while (a != b)
        {
            if (a < b)
                b -= a;
            else
                a -= b;
        }
    }
}
```

Výpis 1: Program v jazyce Java

```
public class MyClass
{
    public int MyMethod(int a, int b)
    {
        while (a != b)
        {
            if (a < b)
                b -= a;
            else
                a -= b;
        }
    }
}
```



Obrázek 1: Pokusný obrázek – absolutní velikost



Obrázek 2: Pokusný obrázek – relativní velikost

}

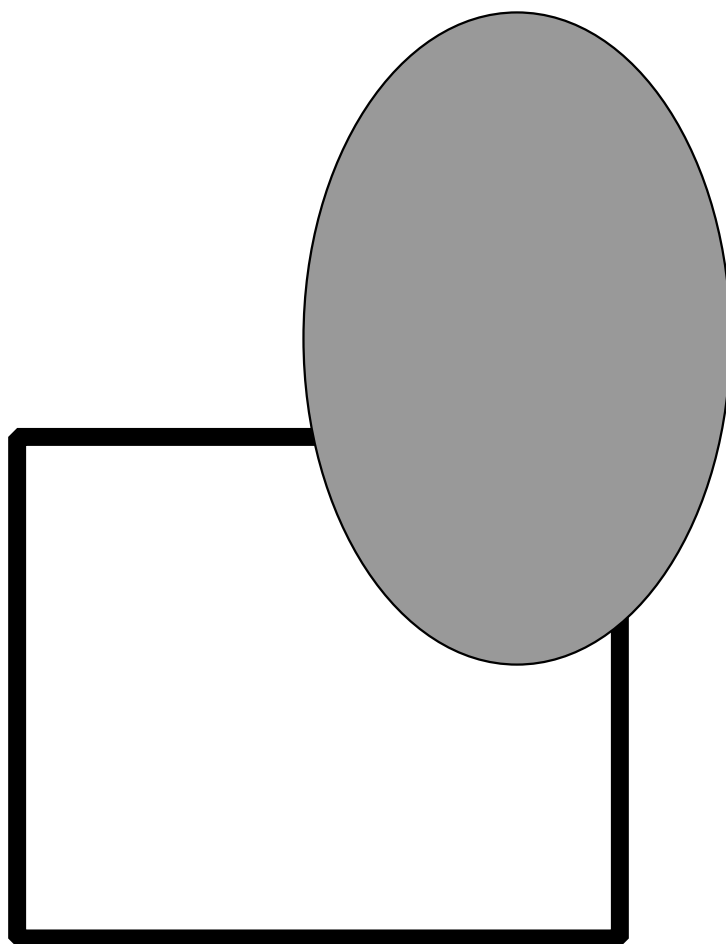
Výpis 2: Program v jazyce Java, načtený z externího souboru

```
procedure X(i : integer; var x : real);  
begin  
  x := i + 3;  
end;
```

Výpis 3: Program v Pascalu

4.4 Obrázky a tabulky

A ještě si můžeme zkusit vysázet obrázek. Obrázek 1 má určenu absolutní velikost, zatímco obrázek 2 je určen relativně vůči šířce textu.



Obrázek 3: Pokusný obrázek – otočený naležato

q	$\delta(q, 0)$	$\delta(q, 1)$
q_0	q_1	q_0
q_1	q_1	q_2
q_2	q_1	q_0

Tabulka 1: Pokusná tabulka

A ještě zkusíme vysázet několik tabulek, ale jen kvůli seznamu tabulek v úvodu. Tabulka 1 představuje jednoduchou tabulku, která se svou šířkou pohodlně vejde do šířky textu. Velké tabulky, stejně jako obrázky, můžeme vysázet naležato. Ukázkou velké, komplikované tabulky¹ je tabulka 2.

¹Pokud, ale píšete práci česky, měly by být tabulky také česky – mě se jen nechtěla předělávat do češtiny.

File	bible.txt	world.txt	law.txt	latimes.txt
Language Format	English Plain text	English Plain text	Czech Plain text	English SGML
Size of file [bytes]	4047392	2473400	64573143	498360166
Number of tokens	1532262	684767	19432898	161254928
Number of words	766131	342383	9716449	70766067
Number of nonwords	766131	342384	9716449	80619289
Number of controls				9869572
				6.12%
Number of unique tokens	13791	23564	250570	529482
Number of unique words	13744	23082	246266	524280
Number of unique nonwords	47	482	4304	3079
Number of unique controls				2123
				0.582%
				0.401%
Word average frequency	55.743	14.833	39.455	134.978
Nonword average frequency	16300.66	710.34	2257.539	26183.595
Control average frequency				4648.88
Minimal length of word	1	1	1	1
Maximal length of word	18	27	41	58
Minimal length of nonword	1	1	1	1
Maximal length of nonword	4	56	700	253
Minimal length of control				3
Maximal length of control				132

Tabulka 2: Experimental Files — Detailed Statistics

5 Závěr

Tak doufám, že Vám tato ukázka k něčemu byla. Další informace najdete v publikacích [1, 2].

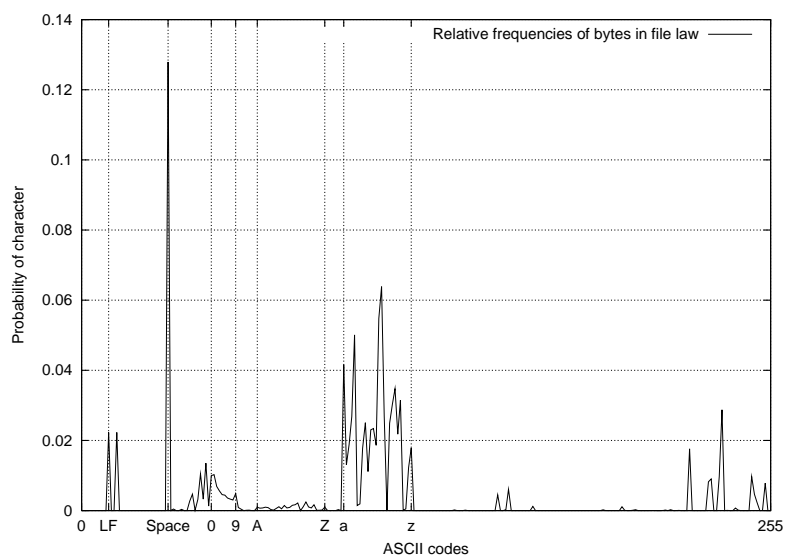
Jiří Dvorský

6 Reference

- [1] Goossens, Michel, *The L^AT_EX companion*, New York: Addison, 1994.
- [2] Lamport, Leslie, *L^AT_EX: a document preparation system: user's guide and reference manual*, New York: Addison-Wesley Pub. Co., 1994.

A Grafy a měření

Tohle je příloha k práci. Většinou se sem dávají grafy, tabulky, které by vzhledem ke svému počtu překážely v textu diplomky.



Obrázek 4: Nějaký graf