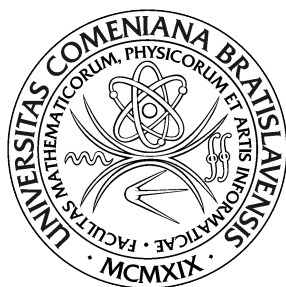


UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



# VERIFIKAČNÝ NÁSTROJ PRE UNITY

Diplomová práca

2017

Bc. Filip Špaldon

UNIVERZITA KOMENSKÉHO V BRATISLAVE  
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



# VERIFIKAČNÝ NÁSTROJ PRE UNITY

Diplomová práca

Študijný program: Aplikovaná informatika  
Študijný odbor: 2511 Aplikovaná informatika  
Školiace pracovisko: Katedra aplikovanej informatiky  
Školiteľ: doc. RNDr. Damas Gruska, PhD.

Bratislava, 2017

Bc. Filip Špaldon

Čestne prehlasujem, že túto diplomovú prácu som  
vypracoval samostatne len s použitím uvedenej literatúry  
a za pomoci konzultácií u môjho školiteľa.

Bratislava, 2017

.....

Bc. Filip Špaldon

# Obsah

<b>1</b>	<b>Východiská</b>	<b>1</b>
1.1	UNITY . . . . .	1
1.2	Vlastnosti UNITY . . . . .	1
1.2.1	Nedeterminizmus . . . . .	1
1.2.2	Absencia toku riadenia (control-flow) . . . . .	2
1.2.3	Synchrónnosť a asynchrónnosť . . . . .	2
1.2.4	Stavy a priradenia . . . . .	2
1.3	Telo programu . . . . .	2
1.3.1	Declare-section . . . . .	3
1.3.2	Always-section . . . . .	3
1.3.3	Initially-section . . . . .	4
1.3.4	Assign-section . . . . .	5
1.3.5	Ukážka programu . . . . .	5
1.4	Syntaktický strom . . . . .	5
1.5	Syntaktická analýza . . . . .	5
1.6	Model checking . . . . .	5
1.7	LTSmin . . . . .	5
1.7.1	Back-ends . . . . .	5
1.7.2	Front-ends . . . . .	5

# Kapitola 1

## Východiská

### 1.1 UNITY

Unity vychádza z knihy Parallel Program Design - A Foundation, v ktorej bol Unity popísaný a navrhnutý autormi K. Mali Chandy a Jayadev Misra z Univerzity of Texas.

### 1.2 Vlastnosti UNITY

- Nedeterminizmus
- Absencia toku riadenia (control-flow)
- Synchronnosť a asynchronnosť
- Stavy a priradenia

#### 1.2.1 Nedeterminizmus

...

### 1.2.2 Absencia toku riadenia (control-flow)

...

### 1.2.3 Synchronnosť a asynchrónnosť

...

### 1.2.4 Stavý a priradenia

...

## 1.3 Telo programu

Unity obsahuje štyri základné sekcie: deklaráciu premenných, množinu skratiek, počiatočné hodnoty premenných a množinu priradovacích príkazov. V tele programu sa tieto sekcie vyskytujú pod názvami `declare`, `always`, `initially`, `assign`. Telo programu obsahuje aj `program-name`, názov programu, ktorý môžeme vynechať, v tom prípade z tela programu vynechávame aj sekciu `program-name`. Unity program má nasledujúcu formu:

<b>Program</b>	<i><b>program-name</b></i>
<b>declare</b>	<i><b>declare-section</b></i>
<b>always</b>	<i><b>always-section</b></i>
<b>initially</b>	<i><b>initially-section</b></i>
<b>assign</b>	<i><b>assign-section</b></i>
<b>end</b>	

Obr. 1.1: Ukážka tela programu

### 1.3.1 Declare-section

Táto sekcia obsahuje deklaráciu premenných použité v programe a ich súvisiace typy. V nasledujúcej ukážke môžete vidieť deklaráciu premenných `x` a `y` typu `integer`. Syntax je podobná ako v programovacom jazyku PASCAL.

Medzi základné typy patria:

- Integer
- Boolean

Príklad deklarácie:

---

```
declare  
  x, y : integer
```

---

Taktiež sa využívajú `n`-rozmerné polia v nasledujúcom tvare:

---

```
declare  
  p: Array[a1, a2, ..., an] of integer
```

---

### 1.3.2 Always-section

Sekcia `always` definuje skratky, ktoré slúžia na stručné spísanie programu. Konkrétnejšie to sú premenné, ktoré definujú funkcie alebo podmienky. Takéto premenné sú známe ako transparentné premenné. Transparentné premenné poskytujú vhodný spôsob skrátenia výrazov, ktoré sa často vyskytujú v programe. Táto sekcia nie je nevyhnutná v tele programu Unity.

Transparentné premenné môžeme definovať následovne pomocou `||`:

---

```
always
    decx = x > y
||
    decy = y > z
```

---

Tieto premenné je možné zapísať aj jednoriadkovo bez použitia spojovníka:

---

```
always
    decx, decy = z > y, y > z
```

---

### 1.3.3 Initially-section

Initially sekcia je súbor rovníc, ktoré definujú počiatočné hodnoty pre niektoré programové premenné. Premenné, ktoré nie sú inicializované majú ľubovoľné počiatočné hodnoty. Premenné  $x$  a  $y$  môže byť definované:

---

```
initially
    x = X
||
    y = Y
```

---

alebo takto:

---

```
initially
    x, y = X, Y
```

---



### 1.3.4 Assign-section

### 1.3.5 Ukážka programu

Nasledujúci Unity program predsavuje Euclidovský algoritmus pre nájdenie najväčšieho spoločného deliteľa čísel  $X, Y$ :

<b>Program</b>	<i>GCD</i>
<b>declare</b>	$x, y : \textit{natural}$
<b>initially</b>	$x, y = X, Y$
<b>assign</b>	$x := x - y \text{ if } x > y$
	$y := y - x \text{ if } y > x$
	‖
<b>end</b>	

Obr. 1.2: GCD algoritmus

## 1.4 Syntaktický strom

## 1.5 Syntaktická analýza

## 1.6 Model checking

## 1.7 LTSmin

### 1.7.1 Back-ends

### 1.7.2 Front-ends

# Literatúra

- [1] A UNITY - Style Programming Logic for a Shared Dataspace Language  
- H. Conrad Cunningham and Gruia-Catalin Roman - 1989
- [2] Mechanizing UNITY in Isabelle - Lawrence C. Paulson - 2000

# Zoznam obrázkov

1.1	Ukážka tela programu . . . . .	2
1.2	GCD algoritmus . . . . .	5