

**S T U • •**  
**• • • • •**  
**• F E I •**  
**• • • • •**

## **Zadanie č. 2**

Automaty a formálne jazyky

**Autor:** Bc. Martin Kiesel

# Obsah

<b>1</b>	<b>Zadanie</b>	<b>3</b>
<b>2</b>	<b>Riešenie</b>	<b>4</b>
	Vývojová špecifikácia . . . . .	4
	Používanie AFJ-assignment-2 . . . . .	4
	AFJ-assignment-2 (BONUS) . . . . .	4
	Obmedzenia AFJ-assignment-2 . . . . .	5
<b>3</b>	<b>Záver</b>	<b>5</b>

## 1 Zadanie

Vo zvolenom (podľa seba) programovacom jazyku nájdite vhodnú reprezentáciu konečného automatu a zostavte program, ktorý:

pre zadaný DKA a zadané slovo simulovaním výpočtu určí, či slovo patrí do jazyka akceptovaného zadaným DKA alebo nie;

k zadanému NKA nájde ekvivalentný DKA;

k zadanému DKA nájde ekvivalentný redukovaný (minimálny) automat.

## 2 Riešenie

Riešením zadania je konzolový program naprogramovaný v C++.

### Vývojová špecifikácia

AFJ-assignment-2 je vyvíjaný na operačnom systéme OS X, vo vývojovom prostredí Xcode (Version 7.2.1) v programovacom jazyku C++ s dialektom GNU++11. Systém riadenia revízií (SCM) je Git 2.7.2 a repozitár projektu je hostovaný na [GitHub-e](#). Dokumentácia je písaná v  $\text{\LaTeX}$  s online kompilátorom [sharelatex](#). Všetky názvy metód, premenných a chybové hlásenia sú v anglickom jazyku.

### Používanie AFJ-assignment-2

AFJ-assignment-2 (ďalej len program) používa getopt na parsovanie argumentov. Program akceptuje štyri argumenty a z toho tri s povinným parametrom. Bezparametrový argument je `-h` a argumenty s povinným parametrom sú `-i`, `-o`, `-w`. Argumenty môžu byť napísané v akomkoľvek poradí. Vstupné aj výstupné súbory sú v tvare XML, ktorým rozumie program jFlap 7.0.

Vysvetlenie argumentov:

- h vypíše nápovedu program
- i [INPUT FILE] vstupný súbor (jff-xml formát)
- o [OUTPUT FILE] výstupný súbor (jff-xml formát)
- w [WORD] vstupné slovo (epsilon znak je medzera ' ')

Program po spustení vypočíta, či vstupný (-i) automat je deterministický alebo nedeterministický:

- ak je vstupný automat nedeterministický program ho prekonvertuje na deterministický uloží ho do výstupného súboru a ukončí sa
- ak je vstupný automat deterministický a vstupné slovo (-w) nie je definované, vykoná minimalizáciu deterministického automatu uloží ho do výstupného súboru a ukončí sa
- ak je vstupný automat deterministický a vstupné slovo je definované, program vykoná simulovaný výpočet a určí či vstupné slovo patrí do jazyka akceptovaného vstupným automatom

### AFJ-assignment-2 (BONUS)

AFJ-assignment-2 program pracuje s formátom XML (jFlap), pred každým uložením automatu do XML súboru, program vypočíta súradnice pre každý stav, pričom výsledný automat v programe jFlap je v kruhovom usporiadaní (circle layout).

## Obmedzenia AFJ-assignment-2

Program nie je idiot-proof, a teda číta iba jFlap súbory. Pri minimalizácii DKA program neodstraňuje mŕtve stavy (dead state).

## 3 Záver

Výsledkom zadania je funkčný program, ktorý spĺňa všetky požiadavky zadania. Zdrojový kód tohto zadania je voľne dostupný na serveroch [GitHub-u](#).

Cieľ zadania sa mi podarilo splniť.

*27. Marec 2016  
8787 – Martin Kiesel*