

Dalyko kodas	Kreditai
INF4001	6

Dalyko pavadinimas lietuvių kalba

FORMALIOSIOS KALBOS IR JŲ TAIKYMAI

Dalyko pavadinimas anglų kalba

FORMAL LANGUAGES AND APPLICATIONS

Trumpa dalyko anotacija lietuvių kalba (iki 500 simbolių)

Dalyko tikslas – supažindinti su pagrindinėmis formaliųjų kalbų sąvokomis, baigtinių bei steko tipo automatų savybėmis ir programavimo kalbų transliavimo metodais. Studentai išmoksta sudaryti deterministinius ir nedeterministinius baigtinius automatus, reguliariąsias išraiškas, steko tipo automatus, nuo konteksto nepriklausančias gramatikas, transformuoti tarpusavyje baigtinius automatus ir reguliariąsias išraiškas, ir transformuoti tarpusavyje steko tipo automatus ir nuo konteksto nepriklausančias gramatikas. Įsisavinami pagrindiniai leksinės ir sintaksinės analizės principai, leksinių skenerių ir sintaksinių analizatorių generatoriai JLex ir JCUP, pagrindiniai kodo generavimo ir optimizavimo principai. Įgyjama grupinio darbo patirtis sudarant paprastos nuo konteksto nepriklausančios kalbos kompiliatorių.

Dalyko anotacija anglų kalba (iki 500 simbolių)

This course is an introduction to formal languages and compiler design. During the course, students will gain knowledge on finite automata, regular and context-free languages and grammars, pushdown automata, lexical analysis, parsing, compiler compilers, semantic analysis, code generation and optimization as well as gain practical experience in compiler construction. The course structure consists of lectures, laboratory works, and individual work.

Būtinasis pasirėngimas dalyko studijoms

Diskrečios struktūros ir matematinė logika (INF1006). Programavimo pagrindai.

Dalyko tikslas

Supažindinti su pagrindinėmis formaliųjų kalbų klasėmis, gramatikomis ir automatais.

Dalyko turinys

Nr.	Turinys (temos)
1.	Automatai. Formalusis įrodymas; Papildomos įrodymo schemas; Įrodymas indukcijos metodu; Pagrindinės automatų teorijos sąvokos.
2.	Baigtinis automatas. Deterministiniai baigtiniai automatai (DBA); Nedeterministiniai baigtiniai automatai (NDBA); Baigtiniai automatai su ϵ -perėjimais.
3.	Reguliariosios išraiškos ir kalbos. Reguliariosios išraiškos; Baigtiniai automatai ir reguliariosios išraiškos; Reguliariųjų išraiškų taikymai; Algebriniai reguliariųjų išraiškų dėsniai.
4.	Reguliariųjų kalbų savybės. Kalbų nereguliarumo įrodymas; Uždarumas; Išsprendžiamumas; Automatų ekvivalentiškumas. Automatų minimizavimas.
5.	Nuo konteksto nepriklausančios gramatikos ir kalbos. Nuo konteksto nepriklausančios gramatikos (NKNG); Analizės medžiai; Nuo konteksto nepriklausančių gramatikų taikymai; Gramatikų ir kalbų dviprasmiškumas.
6.	Steko tipo automatai. Steko tipo automato (STA) apibrėžimas; STA priimamos kalbos; STA ir nuo konteksto nepriklausančių kalbų (NKNK) ryšys; Deterministinis steko tipo automatas.
7.	Nuo konteksto nepriklausančių kalbų savybės. Normaliosios NKNG formos; Nuo konteksto nepriklausančių kalbų kaupinimo lema; NKNK uždarumo savybės; NKNK išsprendžiamumo savybės.
8.	Programavimo kalbų transliavimas ir kompiliatoriaus struktūra. Pagrindinių programavimo paradigmu apžvalga; Programavimo kalbų konstravimas; Programų kompiliavimas ir interpretavimas.
9.	Leksinė analizė. Leksinis skeneris; Lex; Simbolių lentelė.
10.	Sintaksinė analizė. BNF ir EBNF; Deterministinė sintaksinė analizė, FIRST ir FOLLOW, LL ir LR gramatikos.
11.	Deterministinė smulkinamoji sintaksinė analizė. Kairinės rekursijos pašalinimas; Kairinis skaidymas; Prognozuojančioji sintaksinė analizė; Rekursinis nusileidimas.
12.	Deterministinė stambinamoji sintaksinė analizė. Pirmenybinė analizė, LR analizė; Yacc.
13.	Aprašai, moduliai ir atminties valdymas. Aprašų modeliai; Parametrizavimo mechanizmai; Parametrizuoti tipai; Aprašų matomumo bendrinimo ir ribojimo mechanizmai; Šiukšlių rinkimas.
14.	Kodo generavimas. Tarpinis ir objektinis kodas; Tarpinis įvaizdis; Programos kodo generatorių realizacija; Programos kodo generavimas apeinant medį; Kontekstinis transliavimas; Registras.
15.	Optimizavimas. Nuo konkretaus kompiuterio architektūros nepriklausomas optimizavimas; Duomenų srauto analizė; Ciklų optimizavimas; Nuo konkretaus kompiuterio architektūros priklausomas optimizavimas.

Studentų darbo krūvio paskirstymas valandomis (kontaktinio ir savarankiško darbo val.)

Paskaitos (P)	45 val.
Laboratoriniai darbai (L)	30 val.
Savarankiškas darbas	85 val.
Iš viso	160 val.

Kaupiamojo balo sandara ir jo dedamųjų svoris

Koliokviumas – 17%, Laboratoriniai darbai – 33%, Egzaminas – 50%.

Rekomenduojama literatūra

Nr.	Leidimo metai	Leidinio autoriai ir pavadinimas	Leidykla	Egzempliorių skaičius		
				Universiteto bibliotekoje	Metodiniuose kabinetuose	Kitose bibliotekose
<i>Pagrindinė literatūra</i>						
1.	2011	A.Deveikis. Formaliųjų kalbų praktikumas.	VDU			etalpykla.vdu.lt
2.	2010	T.Æ. Mogensen. Basics of Compiler Design.	DIKU, University of Copenhagen			http://www.diku.dk/~torbenm/Basics
3.	2001	J.E.Hopcroft, R.Motwani and J.D.Ullman. Introduction to Automata Theory, Languages, and Computation 2 nd Ed.	Addison Wesley	1		
<i>Papildoma literatūra</i>						
4.	2004	R.B.Yehezkael. Course Notes on Formal Languages and Compilers.	Jerusalem College of Technology	http://homedir.jct.ac.il/~rafi/		
5.	1986	A.V.Aho, R.Sethi, and J.D.Ullman. Compilers: Principles, Techniques, and Tools.	Addison Wesley	http://cs.uccs.edu/~gsc/pub/phd/ftorres/doc/Compiler.pdf		

Dalyko programos rengėjas/-ai

Doc. Dr. Algirdas Deveikis, Informatikos fakultetas, Taikomosios informatikos katedra