

Nazwa modułu (blok przedmiotów): JĘZYKI FORMALNE I METODY KOMPILACJI		Kod modułu: M21					
Wypełnia Zespół Kierunku	Nazwa przedmiotu: Języki formalne i metody kompilacji		Kod przedmiotu:				
	Nazwa jednostki prowadzącej przedmiot / moduł: INSTYTUT INFORMATYKI STOSOWANEJ						
	Nazwa kierunku: INFORMATYKA						
	Forma studiów: stacjonarne		Profil kształcenia: PRAKTYCZNY		Specjalność:		
	Rok / semestr: 3/5		Status przedmiotu / modułu: obowiązkowy		Język przedmiotu / modułu: polski		
	Forma zajęć	wykład	ćwiczenia	ćwiczenia laboratoryjne	konwersatorium	seminarium	inne (wpisać jakie)
	Wymiar zajęć	15		15			
	Koordynator przedmiotu / modułu		dr Stefan Sokółowski				
Prowadzący zajęcia		dr Stefan Sokółowski					
Cel przedmiotu / modułu		<ol style="list-style-type: none"> 1. Nauczenie opisywania składni w notacji BNF lub podobnej 2. Zapoznanie z prostymi metodami budowania drzew wyvodu danego tekstu dla danej gramatyki 3. Zapoznanie z podstawami programowania w kodzie wewnętrznym komputera 4. Zapoznanie z prostymi metodami generowania kodu wewnętrznego odpowiadającego programowi w języku wysokiego poziomu 5. Zapoznanie ze sposobami działania kompilatorów (bez szczegółów) 					
Wymagania wstępne		<ol style="list-style-type: none"> 1. Znajomość klasycznych metod programowania, umiejętność konstruowania algorytmów 2. Obycie z rekursją 3. Biegłość w programowaniu w języku wysokiego poziomu (aktualnie ilustrowane na przykładzie C, ale konkretny język nie ma większego znaczenia) 					
EFEKTY KSZTAŁCENIA					Odniesienie do efektów dla programu		
Nr	Wiedza						
01	Zna teorię języków formalnych				K_W01		
02	Zna oprogramowanie niskiego poziomu i kompilatory				K_W05 K_W06 K_W09		
	Umiejętności						
03	Programuje fragmenty kompilatorów				K_U15 K_U16 K_U17		
	Kompetencje społeczne						
04	Rozumie istotę publikacji Noama Chomsky'ego				K_K02		
TREŚCI PROGRAMOWE							
Forma zajęć – WYKŁAD							
<ol style="list-style-type: none"> 1. Języki formalne i gramatyki: <ul style="list-style-type: none"> ▪ pojęcie gramatyki i definiowanie języków za pomocą gramatyk; ▪ języki regularne, bezkontekstowe, inne; ▪ problem istnienia gramatyki dla danego języka; ▪ leksyka i składnia języków programowania. 2. Teoretyczne modele urządzeń liczących: <ul style="list-style-type: none"> ▪ automaty skończone; ich związek z językami regularnymi; ▪ automaty ze stosem; ich związek z językami bezkontekstowymi. 3. Analiza leksykalna: <ul style="list-style-type: none"> ▪ pojęcie leksemu; ▪ podział tekstu programu na leksemy za pomocą automatu skończonego; ▪ komputerowa realizacja analizatora leksykalnego, czyli skanera. 2. Analiza składniowa: <ul style="list-style-type: none"> ▪ analiza składniowa, czyli budowa drzewa rozbioru; 							

<ul style="list-style-type: none"> ▪ zstępująca analiza rekursywna; ▪ wstępująca analiza z pierwszeństwem; ▪ komputerowa realizacja analizatora składni czyli parsera.
3. Programowanie w kodzie wewnętrznym maszyny.
4. Analiza semantyczna i generacja kodu: <ul style="list-style-type: none"> ▪ kontekstowe cechy języków programowania; ▪ gospodarka pamięcią; ▪ kompilacja programów z języka wysokiego poziomu na język abstrakcyjnej maszyny stosowej; ▪ kompilacja programów z języka abstrakcyjnej maszyny stosowej na kod wewnętrzny.
5. Narzędzia programistyczne do produkcji kompilatorów: <ul style="list-style-type: none"> ▪ LEX (FLEX) do analizy leksykalnej; ▪ YACC (BISON) do analizy składniowej.
8. Powtórka (1 godz.).
Forma zajęć – LABORATORIUM
1. Wyprowadzanie słów i konstruowanie drzew wyvodu w różnych rodzajach gramatyk (regularnych i bezkontekstowych).
2. Konstruowanie możliwie prostych gramatyk regularnych i bezkontekstowych do zadanych języków.
3. Badanie, czy dane dwie gramatyki definiują ten sam język.
4. Ręczna symulacja działania zadanych automatów skończonych i automatów ze stosem.
5. Konstruowanie automatów skończonych i automatów ze stosem do zadanych problemów i języków.
6. Komputerowa realizacja programów ze stosem (np. obliczanie wartości wyrażeń arytmetycznych).
7. Programowanie prostego wstępującego analizatora składni.
8. Programowanie w języku przypominającym wewnętrzny (na dostarczonym symulatorze).
9. Tłumaczenie prostych programów z języka wysokiego poziomu na język maszyny stosowej.
10. Interpretacja programów maszyny stosowej.

Metody kształcenia	Wykład z prezentacją Zadania praktyczne w laboratorium komputerowym
Metody weryfikacji efektów kształcenia	
	Nr efektu kształcenia z sylabusu
Kolokwium zaliczeniowe (na papierze)	01 02
Sprawdziany komputerowe i kolokwium komputerowe	03
Udział w dyskusji na wykładzie	04
Forma i warunki zaliczenia	Zaliczenie laboratorium: sprawdziany komputerowe i kolokwium. Zaliczenie wykładu: kolokwium.
Literatura podstawowa	<ol style="list-style-type: none"> 1. Kowalski S., Mostowski A.W. Teoria automatów i lingwistyka matematyczna. PWN, 1979. 2. Gries D. Konstrukcja translatorów dla maszyn cyfrowych. WNT, 1984. 3. Salwicki A. Techniki kompilacji. Wykład na Politechnice Białostockiej, slajdy dostępne pod http://aragorn.pb.bialystok.pl/~salwicki/techkomp/techkomp.html 4. Aaby A. Compiler Construction using Flex and Bison. Podręcznik dostępny pod http://cs.wvc.edu/~aabyan/464/Book/. 5. Aho A. V., Ullman J. D. The Theory of Parsing, Translation and Compiling, Vol. 1. Prentice-Hall, 1972. 6. Aho A. V., Sethi R., Ullman J. D. Compilers. Principles, Techniques and Tools. Addison-Wesley, 1986. 7. Blikle A. Automaty i gramatyki. PWN, 1971. 8. Blikle A. Wybrane zagadnienia lingwistyki matematycznej. W: Problemy przetwarzania informacji, Tom 2. (Red. A. Mazurkiewicz), WNT 1974. 9. Hopcroft J. E., Ullman J. D. Wprowadzenie do teorii automatów, języków i obliczeń. PWN, 1994. 10. Waite W. M., Goos G. Konstrukcja kompilatorów. WNT, 1989.
Literatura uzupełniająca	
NAKLAD PRACY STUDENTA:	
	Liczba godzin
Udział w wykładach	15

Samodzielne studiowanie tematyki wykładów	5
Udział w ćwiczeniach audytoryjnych i laboratoryjnych*	15
Samodzielne przygotowywanie się do ćwiczeń*	25
Przygotowanie projektu / eseju / itp. *	
Przygotowanie się do egzaminu / zaliczenia	10
Udział w konsultacjach	5
Inne	
ŁĄCZNY nakład pracy studenta w godz.	75
Liczba punktów ECTS za przedmiot	3 ECTS
Obciążenie studenta związane z zajęciami praktycznymi*	40 1,6 ECTS
Obciążenie studenta na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich	35 1,4 ECTS