**Nazwa przedmiotu:**

Metody translacji

**Koordynator przedmiotu:**

Dr Jan Bródka

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

7 / rok ak. 2015/2016

**Liczba punktów ECTS:**

5

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

1. godziny kontaktowe - 45 h; w tym
a. obecność na wykładach – 30 h
b. obecność na ćwiczeniach – 15 h
2. przygotowanie do ćwiczeń i kolokwium końcowego – 45 h
3. zadania domowe – 10 h
Razem nakład pracy studenta 100 h = 4 pkt. ECTS

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

1. obecność na wykładach – 30 h
2. obecność na ćwiczeniach – 15 h
Razem 45 h, co odpowiada 2 pkt. ECTS

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

1. obecność na ćwiczeniach – 15 h
2. zadania domowe – 10 h
Razem 25 h, co odpowiada 1 pkt. ECTS

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład:  | 30h |
| Ćwiczenia:  | 15h |
| Laboratorium:  | 0h |
| Projekt:  | 0h |
| Lekcje komputerowe:  | 0h |

**Wymagania wstępne:**

Biegła znajomość więcej niż jednego języka programowania wysokiego poziomu, znajomość wyrażeń regularnych i gramatyk bezkontekstowych, znajomość podstawowych struktur danych. Przydatna znajomość języka zorientowanego maszynowo (asemblera)
Przedmioty poprzedzające: Teoria Automatów i Języków, Programowanie, Programowanie Obiektowe, Algorytmy i Struktury Danych I, Programowanie w Językach Zorientowanych Maszynowo (zalecane)

**Limit liczby studentów:**

Bez limitu

**Cel przedmiotu:**

Celem przedmiotu jest zdobycie wiedzy na temat przebiegu procesu kompilacji, zrozumienie wpływu cech języka źródłowego (wysokiego poziomu) na wydajność generowanego kodu maszynowego oraz nabycie umiejętności przetwarzania tekstów o sformalizowanej strukturze metodami stosowanymi w kompilatorach z wykorzystaniem popularnych narzędzi
Po ukończeniu kursu studenci powinni:
znać podstawowe fazy procesu kompilacji (analiza leksykalna, składniowa i semantyczna, generowanie kodu pośredniego i docelowego, optymalizacja) i rozumieć ich znaczenie
znać podstawy teoretyczne procesu analizy kodu źródłowego
rozumieć wpływ różnorodnych konstrukcji języków wysokiego poziomu na wydajność generowanego na ich podstawie kodu wynikowego
rozumieć znaczenie kodu pośredniego i znać podstawowe metody jego optymalizacji
umieć stosować wybrane narzędzia wspomagające analizę leksykalną i składniową

**Treści kształcenia:**

Program wykładu
• Ogólne pojęcia i fazy procesu kompilacji
• Przegląd własności języków programowania
• Modele środowiska wykonawczego
• Analiza leksykalna
• Analiza składniowa i gramatyki bezkontekstowe,
w tym metoda zejść rekurencyjnych, analizatory LL(1), analizatory LR(1)
• Analiza semantyczna i gramatyki atrybutywne
• Diagnostyka i obsługa błędów
• Generowanie kodu pośredniego i wynikowego
• Optymalizacja.

Program ćwiczeń
Ilustracja materiału z wykładu, przykłady użycia generatorów FLEX i Bison oraz innych narzędzi wspomagających tworzenie kompilatorów.

**Metody oceny:**

Pisemne zaliczenie końcowe. Łączną ocenę punktową przelicza się na stopnie według poniższych zasad:
b)  3.5 jeżeli uzyskali od 61 do 70  pkt.
c)  4.0 jeżeli uzyskali od 71 do 80  pkt.
d)  4.5 jeżeli uzyskali  od 81 do 90  pkt.
e)  5.0 jeżeli uzyskali powyżej 90  pkt.

**Egzamin:**

nie

**Literatura:**

Aho A.V., Sethi R., Ullman J. D. Kompilatory: Reguły, metody, narzędzia, WNT 2002
Louden K.C. Compiler Construction: Principles and Practice, PWS 1997
dokumentacja programów FLEX i Bison

**Witryna www przedmiotu:**

brak

**Uwagi:**

## Efekty przedmiotowe

### Profil ogólnoakademicki - wiedza

**Efekt W01:**

Ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie języków i paradygmatów programowania

Weryfikacja:

kolokwium końcowe

**Powiązane efekty kierunkowe:** K\_W06

**Powiązane efekty obszarowe:** T1A\_W03

**Efekt W02:**

Zna podstawowe metody, techniki i narzędzia stosowane przy implementacji języków programowania

Weryfikacja:

kolokwium końcowe

**Powiązane efekty kierunkowe:** K\_W10

**Powiązane efekty obszarowe:** T1A\_W07

### Profil ogólnoakademicki - umiejętności

**Efekt U01:**

Ma umiejętność formułowania algorytmów i ich programowania z użyciem popularnych narzędzi

Weryfikacja:

kolokwium końcowe, zadania domowe

**Powiązane efekty kierunkowe:** K\_U11

**Powiązane efekty obszarowe:** T1A\_U09, T1A\_U14, T1A\_U15