

<b>KARTA OPISU MODUŁU KSZTAŁCENIA</b>		
Nazwa modułu/przedmiotu <b>Programowanie deklaratywne</b>		Kod <b>1010511321010510085</b>
Kierunek studiów <b>Informatyka</b>	Profil kształcenia (ogólnoakademicki, praktyczny) <b>ogólnoakademicki</b>	Rok / Semestr <b>1 / 2</b>
Ścieżka obieralności/specjalność <b>-</b>	Przedmiot oferowany w języku: <b>polski</b>	Kurs (obligatoryjny/obieralny) <b>obligatoryjny</b>
Stopień studiów: <b>I stopień</b>	Forma studiów (stacjonarna/niestacjonarna) <b>stacjonarna</b>	
Godziny Wykłady: <b>15</b> Ćwiczenia: <b>-</b> Laboratoria: <b>15</b> Projekty/seminaria: <b>-</b>		Liczba punktów <b>3</b>
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) <b>kierunkowy</b>		(ogólnouczelniany, z innego kierunku) <b>z danego kierunku</b>
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki <b>nauki techniczne</b>		Podział ECTS (liczba i %) <b>3 100%</b>
<b>Odpowiedzialny za przedmiot / wykładowca:</b>		
Dr inż. A. Michalski email: Artur.Michalski@cs.put.poznan.pl tel. tel. (0-61) 665-2923, fax: (0-61) 877 1525 Instytut Informatyki ul. Piotrowo 2, 60-965 Poznań		
<b>Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:</b>		
1	<b>Wiedza:</b>	Student rozpoczynający ten przedmiot powinien posiadać podstawową wiedzę z logiki matematycznej (obliczeniowej) oraz teorii mnogości.
2	<b>Umiejętności:</b>	Powinien posiadać umiejętność rozwiązywania podstawowych problemów opisanych za pomocą formalnego aparatu matematycznego oraz umiejętność pozyskiwania informacji ze wskazanych źródeł.
3	<b>Kompetencje społeczne</b>	Powinien również rozumieć konieczność poszerzania swoich kompetencji i pogłębiania swoich kwalifikacji w zakresie metod i narzędzi programowania. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.
<b>Cel przedmiotu:</b>		
1. Przekazanie studentom podstawowej wiedzy dotyczącej deklaratywnego paradygmatu programowania (na przykładzie języka Prolog)		
2. Rozwijanie u studentów umiejętności rozwiązywania prostych problemów z wykorzystaniem języka programowania wysokiego poziomu opartego na logice obliczeniowej		
3. Kształtowanie u studentów świadomości różnic między paradygmatem deklaratywnym a pozostałymi paradygmatami programowania		
<b>Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia</b>		
<b>Wiedza:</b>		
1. ma uporządkowaną, podbudowaną teoretycznie wiedzę ogólną w zakresie algorytmów i złożoności, języków i paradygmatów programowania - [K_W4]		
2. ma szczegółową wiedzę nt. algorytmiki - [K_W5]		
3. zna podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów i problemów, implementacji języków programowania - [K_W8]		
4. posiada wiedzę podstawową w zakresie analizy i opisu funkcjonowania automatycznych mechanizmów wnioskowania w językach formalnych opartych na logice obliczeniowej - [K_W8]		
5. dysponuje wiedzą elementarną w zakresie form i metod metaprogramowania, przetwarzania symbolicznych form reprezentacji danych i wiedzy - [K_W4]		
<b>Umiejętności:</b>		

1. potrafi ocenić przydatność języków, metod i narzędzi służących do rozwiązywania zadań typowych dla informatyki, oraz wskazywać właściwe obszary zastosowań metod i narzędzi programistycznych paradygmatu deklaratywnego - [K\_U20]
2. ma umiejętność formułowania algorytmów i ich programowania z użyciem przynajmniej jednego z popularnych narzędzi - [K\_U22]
3. opracować i zaimplementować rozwiązanie problemu programistycznego w kategoriach paradygmatu deklaratywnego z zastosowaniem prostych i złożonych (rekurencyjnych) struktur danych o charakterze symbolicznym - [K\_U22]

**Kompetencje społeczne:**

1. rozumie potrzebę ciągłego pogłębiania swojej wiedzy i doskonalenia umiejętności w zakresie narzędzi programistycznych i rozwijających się paradygmatów programowania - [K\_K1]

**Sposoby sprawdzenia efektów kształcenia**

Efekty kształcenia przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- a) w zakresie wykładów:
  - na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach oraz ćwiczeń realizowanych przy tablicy.
- b) w zakresie laboratoriów:
  - na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

Sprawdzanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności związanych z realizacją zadań laboratoryjnych poprzez kolokwium na zakończenie semestru
- ocenę wiedzy teoretycznej wykazanej na zaliczeniu pisemnym z wykładu (student nie może korzystać materiałów dydaktycznych) w formie testu wielokrotnego wyboru, składającego się z 20 pytań o łącznej wartości 20 punktów, z których połowa wymagana jest do otrzymania oceny pozytywnej

Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:

- efektywność zastosowania zdobytej wiedzy podczas rozwiązywania zadanych problemów,
- wskazywanie trudności percepcyjnych studentów umożliwiające bieżące doskonalenie procesu dydaktycznego.

**Treści programowe**

Wykład 1: Wprowadzenie do języka Prolog

- Przykład prostego programu w języku Prolog: Relacje pokrewieństwa
- Rozszerzanie programu przez wprowadzanie reguł prologowych
- Rekurencyjna definicja reguły prologowej
- Zasady generowania odpowiedzi na postawione pytania
- Deklaratywna i proceduralna interpretacja programu prologowego

Wykład 2: Budowa składniowa i interpretacja programów prologowych

- Reprezentacja danych w Prologu
- Mechanizm unifikacji
- Formalna interpretacja deklaratywna programu prologowego
- Formalna interpretacja proceduralna programu prologowego
- Przykład interpretacji programu: Problem małpy i banana
- Porządek kazułów prologowych i celów
- Związki języka Prolog z logiką formalną
- Podsumowanie

Wykład 3: Listy, operatory i operacje arytmetyczne

- Reprezentacja list w Prologu
- Wybrane operacje na listach w Prologu
- Notacja operatorów
- Operacje arytmetyczne w Prologu

Wykład 4: Sterowanie mechanizmem nawrotów

- Mechanizm odcięć (ang. cuts)
- Przykłady wykorzystywania odcięć w programie prologowym
- Negacja przez niepowodzenie
- Problemy związane z zastosowaniem odcięć i negacji w Prologu

Wykład 5: Predefiniowane (systemowe) predykaty prologowe - metapredykaty

- Sprawdzanie typu termów
- Kompozycja i dekompozycja termów: =.., functor, arg, name
- Różne rodzaje operacji równości w Prologu
- Manipulacja bazą danych w Prologu
- Manipulowanie przepływem sterowania w Prologu
- Predykaty: bagof, setof i findall

Wykład 6: Operacje wejścia/wyjścia w Prologu

- Operacje na plikach
- Przetwarzanie plików termów
- Manipulowanie danymi znakowymi
- Kompozycja i dekompozycja atomów
- Wczytywanie programów prologowych: consult i reconsult

Wykład 7: Styl i technika programowania w Prologu

- Ogólne zasady poprawnego programowania w Prologu
- Jak interpretować programy prologowe?
- Styl programowania
- Efektywność programów prologowych

Program ćwiczeń laboratoryjnych odpowiada tematycznie programowi wykładu.

Metody dydaktyczne:

1. wykład: prezentacja multimedialna w tym także przykłady zadań
2. ćwiczenia laboratoryjne: realizacja praktycznych zadań o rosnącym stopniu trudności

<b>Literatura podstawowa:</b>		
1. Prolog. Programowanie, W.F. Clocksin, C.S. Mellish, Helion, Gliwice, 2003		
2. Prolog - programming for AI, I. Bratko, Addison-Wesley, 1990		
3. Logika w rozwiązywaniu zadań, R.A. Kowalski, WNT, Warszawa, 1989		
<b>Literatura uzupełniająca:</b>		
1. Programowanie z bazą wiedzy, E.C. Tyugu, WNT, Warszawa		
2. Prolog, F. Kluźniak, S. Szpakowicz, WNT, Warszawa, 1983		
3. Micro-Prolog, K.L. Clark, F.G. McGabe, WNT, Warszawa, 1985		
<b>Bilans nakładu pracy przeciętnego studenta</b>		
<b>Czynność</b>		<b>Czas (godz.)</b>
1. udział w zajęciach laboratoryjnych: 15 x 1 godz.,		15
2. przygotowanie do ćwiczeń laboratoryjnych: 15 x 0,5 godz.,		7
3. przygotowanie do kolokwium		15
4. udział w wykładach		15
5. przygotowanie do zaliczenia wykładów i udział w kolokwium zaliczeniowym		15
6. udział w konsultacjach związanych z realizacją procesu kształcenia, w szczególności ćwiczeń laboratoryjnych		4
<b>Obciążenie pracą studenta</b>		
<b>forma aktywności</b>	<b>godzin</b>	<b>ECTS</b>
Łączny nakład pracy	69	3
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	34	2
Zajęcia o charakterze praktycznym	15	1