



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Podstawy programowania - Delphi

Przedmiot

Kierunek studiów
informatyka

Studia w zakresie (specjalność)

-

Poziom studiów

pierwszego stopnia

Forma studiów

niestacjonarne

Rok/semestr

1/1

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

Liczba godzin

Wykład

12

Laboratoria

20

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

5

Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

prof. dr hab. Andrzej Marciniak

e-mail: Andrzej.Marciniak@put.poznan.pl

tel.: 61 665-2984

Instytut Informatyki PP

ul. Piotrowo 2, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Andrzej Urbański

e-mail: Andrzej.Urbanski@cs.put.poznan.pl

tel.: 61 665-2984

Instytut Informatyki PP

ul. Piotrowo 2, 60-965 Poznań

Wymagania wstępne

Zgodnie z podstawą programową kształcenia ogólnego dostępną na stronie

http://bip.men.gov.pl/men/bip/akty_prawne/rozporzadzenie_20081223_zal_4.pdf zakłada się, że rozpoczynając przedmiot student ma podstawową wiedzę i umiejętności:

- z matematyki - IV etap edukacyjny, zakres podstawowy poszerzony o rachunek różniczkowy (z zakresu rozszerzonego),

- z informatyki - IV etap edukacyjny, zakres podstawowy.



Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy, jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista i szacunek dla innych ludzi.

Cel przedmiotu

Celem przedmiotu jest zapoznanie studenta z podstawami programowania komputerów oraz nauczenie programowania w języku Delphi (dawniej Pascal). W szczególności obejmuje to:

- przekazanie studentom podstawowych informacji o rozwoju języków programowania, programowaniu strukturalnym, zasadach programowania zorientowanego obiektowo oraz konstrukcji programów tekstowych i okienkowych,
- rozwijanie u studentów umiejętności algorytmizacji problemów i ich oprogramowaniu, w tym w postaci funkcji i procedur,
- nauczenie studentów biegłego posługiwania się zintegrowanym systemem programowania,
- opanowanie przez studentów techniki programowania zorientowanego obiektowo, w tym tworzenia różnych jednostek programowych oraz dostępu do danych i kodów w nich zawartych,
- nauczenie studentów obsługi komponentów oraz wykorzystania ich w programach,
- nabycie przez studentów umiejętności programowego zabezpieczania kodów przed błędami wykonywania programów.

Przedmiotowe efekty uczenia się

Wiedza

1. Ma rozszerzoną i pogłębioną wiedzę z matematyki przydatną do formułowania i rozwiązywania złożonych zadań informatycznych dotyczących m. in. programowania w logice, formalnej specyfikacji i weryfikacji oprogramowania [K1st_W1].
2. Ma uporządkowaną i podbudowaną teoretycznie wiedzę ogólną w zakresie kluczowych zagadnień informatyki oraz wiedzę szczegółową w zakresie wybranych zagadnień tej dyscypliny [K1st_W4].
3. Zna podstawowe techniki, metody oraz narzędzia wykorzystywane w procesie rozwiązywania zadań informatycznych, głównie o charakterze inżynierskim [K1st_W7].

Umiejętności

1. Potrafi, formułując i rozwiązując zadania informatyczne, zastosować odpowiednio dobrane metody, w tym metody analityczne, symulacyjne lub eksperymentalne [K1st_U4].
2. Ma umiejętności formułowania algorytmów i ich implementacji z użyciem przynajmniej jednego z popularnych narzędzi [K1st_U11].
3. Potrafi organizować, współdziałać i pracować w grupie, przyjmując w niej różne role oraz potrafi odpowiednio określić priorytety służące realizacji określonego przez siebie lub innych zadania [K1st_U18].



Kompetencje społeczne

Ma świadomość znaczenia wiedzy w rozwiązywaniu problemów inżynierskich oraz zna przykłady i rozumie przyczyny wadliwie działających systemów informatycznych.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

- na podstawie odpowiedzi udzielanych odnośnie realizacji zadań w ramach ćwiczeń.

Ocena podsumowująca

Sprawdzanie założonych efektów kształcenia jest realizowane przez:

- ocenę umiejętności związanych z realizacją zadań laboratoryjnych (w ciągu semestru każdy student ma do napisania cztery programy o stopniowo wzrastającej skali trudności),
- ocenę wiedzy i umiejętności wykazanych na pisemnym kolokwium zaliczeniowym (po ostatnim wykładzie) o charakterze problemowym i praktycznym (kolokwium składa się z 6 tematów/zadań o różnej skali trudności i wynikającej stąd różnej punktacji; maksymalnie można uzyskać 20 punktów; na ocenę 3.0 trzeba zdobyć 11 punktów).

Treści programowe

Program przedmiotu obejmuje następujące zagadnienia:

- podstawowe pojęcia związane z programowaniem (programowanie, algorytm, program, język programowania, język ukierunkowany maszynowo, rozkaz, język wyższego rzędu, język uniwersalny, język specjalizowany),
- przegląd języków programowania (Ada, Algol, asemblery, Basic, C, C++, Cobol, Fortran, Java, Lisp, Logo, Pascal, PHP, PL/1, Prolog, Python, Ruby),
- sieci działań (schematy blokowe) i symbole stosowane w nich,
- maksymy programistyczne (na podstawie książki D. Van Tassela "Praktyka programowania"),
- ogólne zasady programowania zorientowanego obiektowo (dziedziczność, hermetyczność i polimorfizm),
- ogólna charakterystyka pakietu Embarcadero Delphi XE10 (w kolejnych latach będą przedstawiane następne wersje wraz z ich zakupem do laboraoriów),
- podstawowe pojęcia związane z konstruowaniem programów w zintegrowanym środowiskiem programowania Delphi (projekt, formatka, komponent, własność, zdarzenie),
- posługiwanie się zintegrowanym środowiskiem programowania Delphi,



- przegląd konstrukcji języka Delphi (program, moduł, biblioteka, funkcje i procedury, klasy i obiekty, typy danych, zmienne, instrukcje),
- struktura programu i modułu,
- podstawowe elementy języka (symbole podstawowe, słowa kluczowe i dyrektywy języka, identyfikatory, liczby, łańcuchy, w tym łańcuchy znaków Unicode, literały logiczne, komentarze i separatory),
- typy danych i ich opis (definiowanie typów, typy proste, łańcuchowe, strukturalne, opisujące obiekty, wskaźnikowe, zgodność typów),
- zmienne (deklaracje zmiennych, zmienne proste, indeksowane, rekordowe, obiektowe, dynamiczne, proceduralne, wariantowe, z początkową wartością, nakładanie zmiennych, literały stałe i zmienne),
- wyrażenia (rodzaje operatorów i ich priorytet, składnia wyrażenia, wyrażenie stałe),
- instrukcje (proste, strukturalne, assemblerowa),
- funkcje i procedury (definicje, rodzaje parametrów, przeciążanie, wywoływanie, konwencje wywoływania),
- przetwarzanie obiektów (konstruktory i destruktory, metody statyczne, wirtualne, dynamiczne i abstrakcyjne, obsługa wiadomości, własności),
- biblioteki łączone dynamicznie (tworzenie bibliotek, statyczne i dynamiczne pobieranie funkcji i procedur z bibliotek),
- pakiety,
- przetwarzanie plików,
- okienka z komunikatami.

Na zajęciach laboratoryjnych studenci, po zapoznaniu się ze zintegrowanym środowiskiem programowania Delphi, piszą programy wykorzystujące poznane elementy języka.

Metody dydaktyczne

1. Wykład - prezentacja multimedialna (każdy wykład) oraz prezentacja pisania i wykonania wybranych programów bezpośrednio w zintegrowanym środowisku programowania Delphi.
2. Zajęcia laboratoryjne - ćwiczenia praktyczne dotyczące elementów języka Delphi, pisanie programów okienkowych w tym języku.

Literatura

Podstawowa

A. Marciniak, Język programowania Delphi, Wydawnictwo NAKOM, Poznań



Uzupełniająca

1. X. Pacheco, S. Teixeira, Delphi 7 - vademecum profesjonalisty, tom 1, Wydawnictwo HELION, Gliwice.
2. M. Cantu, Delphi 7 - praktyka programowania, tom 1 i 2, Wydawnictwo MIKOM, Warszawa.
3. A. Marciniak, Borlan Delphi 5 Professional - Object Pascal, Wydawnictwo NAKOM, Poznań.

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	120	5,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	65	3,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu) ¹	55	2,0

¹ niepotrzebne skreślić lub dopisać inne czynności