

Karta przedmiotu

Nazwa i kod przedmiotu	Programming languages, PG_00045303						
Kierunek studiów	Inżynieria danych						
Data rozpoczęcia studiów	październik 2024 r.		Rok akademicki realizacji przedmiotu		2025/2026		
Poziom kształcenia	I stopnia - inżynierskie		Grupa zajęć		Grupa zajęć obowiązkowych z zakresu kierunku studiów Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	2		Język wykładowy		angielski		
Semestr studiów	3		Liczba punktów ECTS		2.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Wydziały Politechniki Gdańskiej -> Wydział Elektroniki, Telekomunikacji i Informatyki -> Katedra Algorytmów i Modelowania Systemów						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr hab. inż. Piotr Mironowicz				
	Prowadzący zajęcia z przedmiotu		dr hab. inż. Piotr Mironowicz				
			mgr inż. Tomasz Goluch				
			dr hab. inż. Jan Daciuk				
			dr Magdalena Godlewska				
		prof. dr hab. inż. Bogdan Wiszniewski					
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	0.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
	Adresy kursu na platformie eNauczanie:						
	Moodle ID: 1442 Programming languages, 25/26 (winter) https://enauczenie.pg.edu.pl/2025/course/view.php?id=1442						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów	Udział w konsultacjach		Praca własna studenta		RAZEM
	Liczba godzin pracy studenta	30	4.0		16.0		50
Cel przedmiotu	Celem przedmiotu jest zapoznanie się z popularnymi paradygmatami programowania i zdobycie umiejętności ich praktycznego wykorzystywania.						
Efekty uczenia się przedmiotu	Efekt kierunkowy		Efekt z przedmiotu		Sposób weryfikacji i oceny efektu		
	[K6_W81] posiada znajomość struktur gramatycznych oraz obszarów leksykalnych niezbędnych do porozumiewania się w języku obcym w zakresie języka ogólnego oraz specjalistycznego związanego z kierunkiem studiów		Zna podstawowe pojęcia związane z paradygmatami programowania, umie nazwać najważniejsze elementy konstrukcji w poszczególnych językach		[SW3] Assessment of knowledge contained in written work and projects		
	[K6_U04] formułuje logiczne rozwiązania złożonych lub nieustrukturyzowanych problemów		Potrafi zaprogramować rozwiązanie zadanego problemu z wykorzystaniem zadanego języka programowania z różnych paradygmatów		[SU2] Assessment of ability to analyse information [SU4] Assessment of ability to use methods and tools		
	[K6_K03] wykazuje się zdolnością do krytycznego i analitycznego myślenia oraz integruje wiedzę z wielu dyscyplin w celu podejmowania efektywnych decyzji		Rozpoznaje paradygmaty programowania, umie dobrać metodologię do problemu programistycznego		[SK5] Assessment of ability to solve problems that arise in practice [SK2] Assessment of progress of work		

1. Programowanie proceduralne.
2. Składnia liniowa. FORTRAN
3. Rekordy aktywacyjne i podprogramy
4. Rekurencyjne wywołanie procedur.
5. Składnia blokowa. Abstrakcja przepływu sterowania.
6. Wiązanie nazw z obiektami. Zasięg wiązania.
7. Parametry wywołania procedury. Tryby przekazu parametrów.
8. Rekordy aktywacyjne dla języków z rekurencją.
9. Wywołania statyczne i dynamiczne. ALGOL. PASCAL.
10. Ograniczenia języków blokowych.
11. Abstrakcja danych i ochrona dostępu.
12. Modularyzacja. MODULA-2. ADA83, ADA95
13. Sytuacje wyjątkowe. Modele obsługi wyjątków.
14. Współprogramy i procedury współbieżne. Rendezvous.
15. Programowanie obiektowe. Obiekty, klasy, hierarchie.
16. Typy dynamiczne. Polimorfizm. SMALLTALK. C++.
17. Rekurencyjne interpretowanie poleceń.
18. Przekształcenia symboliczne. Rekurencja ogonowa.
19. Funkcyjny paradygmat programowania.
20. Haskell. Podstawowe konstrukcje i operacje, tworzenie modułów, wydajność.
21. Tacit programming.
22. LISP. Ogólny zarys.
23. Prolog jako przykład programowania w logice.
24. Definiowanie języków. Klasyfikacja Chomsky'ego, kompilatory.
25. Monady.

Wymagania wstępne i dodatkowe			
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	kolokwia	50.0%	40.0%
	projekt	50.0%	60.0%
Zalecana lista lektur	Podstawowa lista lektur	1. S. Mangano: XSLT receptury, wyd.2, Helion 2007 2. Cincom Smalltalk Downloads, http://www.cincomsmalltalk.com/ 3. SAXON - The XSLT and XQuery Processor, http://saxon.sourceforge.net/ 4. W.F. Clocksin, W.F., Mellish, C.S.: Prolog Programowanie. Helion 2003 5. Ada Programming, http://en.wikibooks.org/wiki/Ada 6. SWI-Prolog downloads, www.swi-prolog.org/download.html 7. ADA Core, the GNAT Pro Company, http://www.adacore.com/home , https://libre.adacore.com/ 8. D. S. Touretzky: Common Lisp: A Gentle Introduction to Symbolic Computation, http://www.cs.cmu.edu/~dst/LispBook/ 9. Z. Huzar, Z. Fryźlewicz, I. Dubielewicz, B. Hnatk: Ada 95, Helion 1998 10. Polski serwis języka Smalltalk, http://www.objectspace.net/	
	Uzupełniająca lista lektur	http://en.wikipedia.org/wiki/Programming_paradigm	
	Adresy eZasobów		
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania			
Zajęcia praktyczne w ramach przedmiotu	Nie dotyczy		

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.