

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

KARTA OPISU PRZEDMIOTU

Wydział	Informatyki		
Kierunek	Informatyka		
Specjalność	Programowanie		
Semestr	III	Program studiów,	2019/2020
Stopień studiów	II	dla którego obowiązuje	
		syllabus	

Nazwa przedmiotu	Programowanie obiektowe w języku u Java			
Kod przedmiotu	POWJUA			
Łączna liczba godzin	30	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	wykład + laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	3 (1+2)			

Prowadzący zajęcia	
Forma prowadzonych zajęć	Wykład
Wymiar zajęć	15 h
Stopień (tytuł) naukowy	
Imię	
Nazwisko	

Prowadzący zajęcia	
Forma prowadzonych zajęć	Laboratorium
Wymiar zajęć	15 h
Stopień (tytuł) naukowy	
Imię	
Nazwisko	

Wymagania wstępne	Znajomość konstrukcji i umiejętność użycia elementów programowania obiektowego.
Założenia i cele przedmiotu	Nauka języka programowania Java i obiektowego paradygmatu programowania. Przedstawienie zasad i dobrych praktyk w tworzeniu aplikacji zorientowanych obiektowo.
Metody dydaktyczne	<ol style="list-style-type: none"> Wykład – w formie tradycyjnej lub prezentacji multimedialnej Laboratorium – w trakcie którego studenci analizują i rozwiązują problemy/zadania

Efekty uczenia się (odniesienie do charakterystyk poziomów Polskiej Ramy Kwalifikacji)	Odniesienie do efektów dla kierunku	Odniesienie do efektów uczenia się wg Polskiej
---	--	---

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

			Ramy Kwalifikacji
WIEDZA – absolwent zna i rozumie:	W01. Zasady działania maszyny wirtualnej języka JAVA. W02. Ideę i znaczenie procesu kompilacji kodu źródłowego programu. W03. Mechanizmy związane z paradygmatem programowania obiektowego: dziedziczenie, polimorfizm, abstrakcja. W04. Architekturę klas języka JAVA. Różnice między kompozycją a agregacją.	K_W06	P7S_WG
UMIEJĘTNOŚCI – absolwent potrafi:	U01. Napisać kompletny program zorientowany obiektowo. U02. Implementować programy złożone z wielu klas, z zastosowaniem mechanizmów programowania obiektowego. U03. Wykorzystywać mechanizm dziedziczenia i polimorfizmu. U04. Stosować interfejsy i klasy abstrakcyjne.	K_U14 K_U19	P7S_UW
KOMPETENCJE SPOŁECZNE – absolwent jest gotów do	K01. Rozumie potrzeby ciągłego samokształcenia się K02. Potrafi pracować w zespole, przyjmując w nim różne role.	K_K03 K_K04	P7S_UU P7S_UO

Treści programowe		
Lp.	Tematyka zajęć	Liczba godzin
Forma zajęć – wykład		
1	Podstawy języka Java, składnia, typy danych, konstrukcje.	2
2	Klasy i obiekty w języku JAVA.	2
3	Dziedziczenie i polimorfizm. Lambdy i strumienie.	2
4	Interfejsy i klasy abstrakcyjne.	2
5	Lombok.	2
6	Pola finalne, modyfikatory dostępu: public, private, protected. Statyczność.	1
7	Koncepcja obiektowości w Javie w porównaniu do innych języków programowania.	1
8	Rozwój języka w kontekście obiektowości w najnowszych jego wydaniach (od 10 do 21 lub nowszy).	1
9	Prezentacja i code review przykładowych programów napisanych wg koncepcji programowania obiektowego.	2
Forma zajęć – laboratorium		
1	Zadania związane z elementami typu: pętle, instrukcje warunkowe, tablice, kolekcje.	2
2	Zadania zawierające klasy i jej składowe (pola, metody), tworzenie obiektów.	2

WROCŁAWSKA WYŻSZA SZKOŁA INFORMATYKI STOSOWANEJ

3	Tworzenie struktur zawierających kompozycję, agregację i dziedziczenie.	2
4	Realizacja zadań z wykorzystaniem interfejsów.	2
5	Zadania z użyciem zewnętrznych bibliotek, np. Lombok.	2
6	Zadania z użyciem lambd oraz strumieni.	1
7	Omówienie i realizacja zadań z użyciem modyfikatorów dostępu.	1
8	Tworzenie i analiza kompleksowego programu zawierającego koncepcje programowania obiektowego.	3

Forma i warunki zaliczenia przedmiotu	Egzamin pisemny z wykładu. Rozwiązanie list zadań i ćwiczeń w ramach laboratorium.	
Metody weryfikacji efektów uczenia się		Nr efektu uczenia się z sylabusu
	Egzamin pisemny	W01–W04
	Ocena zadań i ćwiczeń wykonanych w ramach laboratorium	U01–U04, K01–K02

Literatura podstawowa	<ol style="list-style-type: none"> 1. Dokumentacja języka Java: https://docs.oracle.com/en/java/javase 2. Materiały dydaktyczne na stronie: www.maciejmuras.com
Literatura uzupełniająca	<ol style="list-style-type: none"> 1. J. Bloch, <i>Java. Efektywne programowanie</i>, wyd. 3, Helion, Gliwice 2018.

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	30
Przygotowanie się do zajęć	15
Studiowanie literatury	15
Udział w konsultacjach	5
Przygotowanie projektu / eseju / prezentacji itp.	-
Przygotowanie się do egzaminu / zaliczenia	15
Inne	-
ŁĄCZNY nakład pracy studenta w godz.	80
Liczba punktów ECTS	3