



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Projekt współfinansowany przez
Unię Europejską w ramach
Europejskiego Funduszu
Społecznego

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Nazwa przedmiotu		Kod ECTS	
Programowanie obiektowe (OA)		11.3.2100	
Nazwa jednostki prowadzącej przedmiot			
Instytut Informatyki			
Studia			
wydział	kierunek	poziom	pierwszego stopnia
Wydział Matematyki, Fizyki i Informatyki	Informatyka	forma	stacjonarne
		moduł	wszystkie
		specjalnościowy	wszystkie
		specjalizacja	wszystkie
Nazwisko osoby prowadzącej (osób prowadzących)			
dr Tomasz Borzyszkowski; dr Maciej Dziemiańczuk; dr Robert Fidytek; prof. UG, dr Piotr Arłukowicz; dr Janusz Dybizbański; dr inż. Tomasz Górski; dr Paweł Pączkowski; dr Ekaterina Cichosz			
Formy zajęć, sposób ich realizacji i przypisana im liczba godzin		Liczba punktów ECTS	
Formy zajęć		7	
Wykład, Ćw. laboratoryjne		30 godz wykl, 30 godz ćw. lab., prac a własna	
Sposób realizacji zajęć		studenta 115h	
zajęcia w sali dydaktycznej		Razem 175h	
Liczba godzin			
Wykład: 30 godz., Ćw. laboratoryjne: 30 godz.			
Termin realizacji przedmiotu			
2024/2025 zimowy			
Status przedmiotu		Język wykładowy	
obowiązkowy		polski	
Metody dydaktyczne		Forma i sposób zaliczenia oraz podstawowe kryteria oceny lub wymagania egzaminacyjne	
<ul style="list-style-type: none"> - Projektowanie doświadczeń - Wykonywanie doświadczeń - Wykład z prezentacją multimedialną 		Sposób zaliczenia	
		Egzamin	
		Formy zaliczenia	
		<ul style="list-style-type: none"> - wykonanie pracy zaliczeniowej - projekt lub prezentacja - egzamin pisemny z pytaniami (zadaniami) otwartymi 	
		Podstawowe kryteria oceny	
		próg zaliczenia	składnik oceny końcowej
	projekty	50%	50%
	aktywność na zajęciach	brak	10%
	egzamin	50%	40%
Sposób weryfikacji założonych efektów uczenia się			

zakładany efekt kształcenia	egzamin	kolokwium	projekt	referat	raport	aktywność w dyskusji	obserwacja postawy studenta
Wiedza							
K_W05	X	X					
P_W1	X	X					
Umiejętności							
K_U07							X
K_U10							X
P_U1							X
Kompetencje społeczne							
K_K01						X	X
K_K03						X	X
P_K1						X	X
P_K2						X	X

Określenie przedmiotów wprowadzających wraz z wymogami wstępnymi

A. Wymagania formalne

Zaliczenie z przedmiotów: Wstęp do programowania i Języki programowania.

B. Wymagania wstępne

Umiejętność konstruowania prostych instrukcji w języku C.

Cele kształcenia

Celem przedmiotu jest zapoznanie studentów z zasadami programowania obiektowego i zdarzeniowego na przykładzie języka Java. Nauczenie studentów podstawowych koncepcji związanych z programowaniem obiektowym oraz umiejętności konstruowania programów obiektowych w języku Java.

Treści programowe

- Wprowadzenie pojęć podstawowych:** klasa, obiekt, pole, metoda; pola i metody: statyczne, publiczne oraz prywatne w języku Java; inicjalizacja i usuwanie obiektu oraz mechanizm tzw. zbieracza śmieci; przegląd instrukcji sterujących języka Java i ich porównanie z instrukcjami sterującymi języka C.
- Pakiety klas języka Java – ukrywanie implementacji:** struktura pakietu; importowanie pakietów; ustalanie praw dostępu do składowych pakietu; budowa interfejsów i ich implementacja.
- Dziedziczenie i polimorfizm: dziedziczenie – składnia i zachowanie praw dostępu do dziedziczonych pól i metod;** od abstrakcji do konkretności: klasy abstrakcyjne i finalne; porównanie własności pól finalnych i statycznych; porównanie własności metod przeciążonych i polimorficznych; przykłady wywołań funkcji polimorficznych.
- Przegląd klas implementujących typowe struktury danych:** pojęcie statycznych i dynamicznych struktur danych; przegląd własności i udostępnianych operacji na typie Collection języka Java i typach pochodnych (List, Set, BitSet, Maps, Vector, Stack i Hashtable); klasy implementujące typy wyliczeniowe i iteratory; polimorficzne metody umożliwiające sortowanie elementów przechowywanych w kolekcjach.
- Programowanie z wykorzystaniem wyjątków:** przegląd predefiniowanych wyjątków w języku Java; zasady tworzenia nowych wyjątków; zgłaszanie i wyłapywanie sytuacji wyjątkowych.
- Programowanie z wykorzystaniem wątków:** pojęcie wątku, zasobów współdzielonych i sekcji krytycznej; przegląd metod współdzielenia zasobów w języku Java: blokowanie zasobów i problem zakleszczenia wątków oraz kolejki priorytetowe i problem uczciwości w dostępie do zasobów; przykłady wykorzystania wątków języka Java do implementacji klasycznych problemów dostępu do zasobów krytycznych.
- Przegląd wybranych wzorców projektowych.**

Wykaz literatury

- Eckel B., Thinking in Java – edycja polska. Wydawnictwo HELION, Warszawa, 2010.
- Horstmann C. S., Cornell G., Core Java 2 - Podstawy. Helion, 2010.
- Horstmann C. S., Cornell G., Core Java 2 - Techniki zaawansowane. Helion, 2010.
- Campione M., Walrath K., Java Tutorial. Addison-Wesley, 2000.

Kierunkowe efekty uczenia się	Wiedza
K_W05: ma ogólną wiedzę na temat różnych paradygmatów programowania i języków programowania; szczegółowo zna metody i wzorce projektowania i programowania obiektowego K_U07: potrafi projektować, tworzyć, uruchamiać i testować	P_W1: Student zna podstawy programowania obiektowego oraz podstawowe konstrukcje języka Java (K_W05)
	Umiejętności P_U1: Student potrafi konstruować obiektowe rozwiązania prostych problemów programistycznych przy użyciu współczesnych narzędzi wspomagających

<p>programy przy wykorzystaniu dedykowanych narzędzi oraz adekwatnych wzorców</p> <p>K_U10: potrafi oceniać przydatność paradygmatów i narzędzi programistycznych do rozwiązywania problemów różnego typu</p> <p>K_K01: zna ograniczenia własnej wiedzy i rozumie potrzebę dalszego uczenia się</p> <p>K_K03: potrafi i jest gotów formułować opinie na temat podstawowych zagadnień informatycznych</p>	<p>wytwarzanie oprogramowania (K_U07, K_U10)</p>
	<p>Kompetencje społeczne (postawy)</p> <p>P_K1: Student zdaje sobie sprawę z tempa rozwoju technologicznego i rozumie potrzebę dalszego uczenia się (K_K01)</p> <p>P_K2: Student potrafi formułować opinie na temat zastosowania podstawowych konstrukcji programistycznych (K_K03)</p>
<p>Kontakt</p>	
<p>t.borzyszkowski@ug.edu.pl</p>	