

KARTA OPISU MODUŁU KSZTAŁCENIA		
Nazwa modułu/przedmiotu Platformy programowania		Kod 1010334561010334966
Kierunek studiów Informatyka	Profil kształcenia (ogólnoakademicki, praktyczny) ogólnoakademicki	Rok / Semestr 3 / 6
Ścieżka obieralności/specjalność -	Przedmiot oferowany w języku: polski	Kurs (obligatoryjny/obieralny) obligatoryjny
Stopień studiów: I stopień	Forma studiów (stacjonarna/niestacjonarna) niestacjonarna	
Godziny Wykłady: 16 Ćwiczenia: - Laboratoria: 16 Projekty/seminaria: -		Liczba punktów 4
Status przedmiotu w programie studiów (podstawowy, kierunkowy, inny) inny		(ogólnouczelniany, z innego kierunku) ogólnouczelniany
Obszar(y) kształcenia i dziedzina(y) nauki i sztuki nauki techniczne nauki techniczne		Podział ECTS (liczba i %) 4 100% 4 100%
Odpowiedzialny za przedmiot / wykładowca: dr inż. Michał Ciesielczyk email: Michal.Ciesielczyk@put.poznan.pl tel. 61 647 5988 Wydział Elektryczny ul. Piotrowo 3A 60-965 Poznań		
Wymagania wstępne w zakresie wiedzy, umiejętności, kompetencji społecznych:		
1	Wiedza:	K_W04: ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie podstawowych algorytmów i ich analizy, technik projektowania algorytmów, abstrakcyjnych struktur danych i ich implementacji, problemów obliczeniowo trudnych; K_W08: ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie baz danych oraz hurtowni danych; K_W12: ma uporządkowaną i podbudowaną metodologicznie wiedzę w zakresie inżynierii oprogramowania
2	Umiejętności:	K_U02: potrafi pracować indywidualnie i w zespole; umie oszacować czas potrzebny na realizację zleconego zadania; potrafi opracować i zrealizować harmonogram prac zapewniający dotrzymanie terminów; K_U03: potrafi opracować dokumentację dotyczącą realizacji zadania inżynierskiego i przygotować tekst zawierający omówienie wyników realizacji tego zadania
3	Kompetencje społeczne	K_U04: ma świadomość odpowiedzialności za pracę własną oraz gotowość podporządkowania się zasadom pracy w zespole i ponoszenia odpowiedzialności za wspólnie realizowane zadania
Cel przedmiotu: Zapoznanie słuchaczy z podstawowymi platformami programistycznymi w językach Java oraz Python.		
Efekty kształcenia i odniesienie do kierunkowych efektów kształcenia		
Wiedza:		
1. ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie podst. konstrukcji programistycznych, implementacji algorytmów, paradygmatów i stylów programowania, metod weryfikacji poprawności programów, języków formalnych, kompilatorów, platform - [K_W05]		
2. orientuje się w obecnym stanie oraz najnowszych trendach rozwojowych informatyki - [K_W19]		
3. zna typowe informatyczne technologie inżynierskie - [K_W18]		
Umiejętności:		
1. potrafi posłużyć się środowiskami i platformami programistycznymi do pisania, wykonywania i testowania prostych programów kodowanych w językach programowania imperatywnego, obiektowego i deklaratywnego - [K_U10]		
2. potrafi sformułować wymagania, opracować model obiektowy oraz ocenić prosty system informatyczny, uwzględniając realizowane funkcje i powiązania między elementami składowymi - [K_U16]		
3. potrafi ocenić przydatność rutynowych metod i narzędzi służących do rozwiązywania prostych zadań inżynierskich typowych dla informatyki oraz wybierać i stosować właściwe technologie - [K_U22]		

Kompetencje społeczne:		
1. rozumie potrzebę i zna możliwości ciągłego dokształcania się (studia drugiego i trzeciego stopnia, studia podyplomowe, kursy) ? podnoszenia kompetencji językowych, zawodowych, osobistych i społecznych - [K_K01]		
Sposoby sprawdzenia efektów kształcenia		
Wykład: Kolokwium zaliczeniowe. Laboratorium: bieżąca ocena w trakcie zajęć, projekt.		
Treści programowe		
Wprowadzenie do programowania obiektowego w języku Java. Kolekcje oraz typy generyczne w Javie. Narzędzia do automatyzacji budowania oprogramowania. Dostęp do bazy danych za pomocą interfejsu JDBC/JPA. Wprowadzenie do biblioteki graficznej JavaFX. Wyszukiwanie pełnotekstowe z Apache Lucene. Metodyka Test-driven Development (TDD). Automatyzacja testów na przykładzie JUnit. Wprowadzenie do programowania w języku Python. Dostęp do bazy danych z wykorzystaniem SQLAlchemy.		
Literatura podstawowa:		
<ol style="list-style-type: none"> Oracle (2017). The Java Tutorials. http://docs.oracle.com/javase/tutorial/ Oracle (2017). Java Platform, Standard Edition 8 API Specification. https://docs.oracle.com/javase/8/docs/api/ Oracle (2017). Outline of the Collections Framework. http://docs.oracle.com/javase/8/docs/technotes/guides/collections/reference.html Oracle (2017). Java SE Technologies - Database. http://www.oracle.com/technetwork/java/javase/jdbc/index.html Oracle (2017). JDBC(TM) Database Access. http://docs.oracle.com/javase/tutorial/jdbc/index.html McCandless M., Hatcher E., Gospodnetić O. (2010). Lucene in Action, Second Edition. Rozdział 1. http://www.manning.com/hatcher3/ JUnit (2017). JUnit. http://www.junit.org/ Python Software Foundation (2017). Welcome to Python. https://www.python.org/ Python Software Foundation (2017). Python 3.x documentation. https://docs.python.org/3/ 		
Literatura uzupełniająca:		
<ol style="list-style-type: none"> Risberg T. (2017). Spring Data JDBC Extensions Reference Documentation. http://docs.spring.io/spring-data/data-jdbc/docs/current/reference/pdf/spring-data-jdbc-ext-reference.pdf Srinivasan K. (2007). Introduction to Java Persistence API(JPA). http://javabeat.net/jpa/ The Apache Software Foundation (2017). Apache Lucene. http://lucene.apache.org/ Chin S. (2017). JavaFX: Making it Easier to Build Better RIAs. https://dzone.com/refcardz/getting-started-javafx Oracle (2017). Writing JUnit Tests in NetBeans IDE. https://netbeans.org/kb/docs/java/junit-intro.html LearnPython.org (2017). Free Interactive Python Tutorial. http://www.learnpython.org/pl/ SQLAlchemy (2017). Object Relational Tutorial. http://docs.sqlalchemy.org/en/rel_0_9/orm/tutorial.html 		
Bilans nakładu pracy przeciętnego studenta		
Czynność	Czas (godz.)	
1. Wykład	16	
2. Laboratoria	16	
3. Przygotowanie do laboratoriów	32	
4. Samodzielna praca na tematy poruszane na wykładzie	16	
5. Konsultacje	2	
Obciążenie pracą studenta		
forma aktywności	godzin	ECTS
Łączny nakład pracy	82	4
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	34	2
Zajęcia o charakterze praktycznym	48	2