



**POLITECHNIKA
BYDGOSKA**

Wydział Telekomunikacji,
Informatyki i Elektrotechniki

Karta przedmiotu
Skryptowe języki programowania

1. Informacje podstawowe

Kierunek studiów informatyka stosowana Jednostka zarządzająca kierunkiem studiów Wydział Telekomunikacji, Informatyki i Elektrotechniki Poziom studiów pierwszego stopnia (inż.) Profil studiów Profil ogólnoakademicki Forma studiów studia stacjonarne		Cykl kształcenia (nabór) 2024/25 Kod przedmiotu 05ISTS.PI4C.1472.24 Języki wykładowe polski Obligatoryjność Obowiązkowy Blok zajęciowy Przedmioty kierunkowe	
Wymagania wstępne	brak wymagań		
Przedmioty wprowadzające	brak przedmiotów wprowadzających		
Koordinator	Tomasz Marciniak		
Okres Semestr 3	Forma zaliczenia Zaliczenie na ocenę Forma prowadzenia i godziny zajęć Wykład: 30 Ćwiczenia laboratoryjne: 30		Liczba punktów ECTS 5.0

2. Efekty uczenia się dla przedmiotu

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
Wiedza:			
W1	Ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie podstaw programowania w językach skryptowych Python, Perl i JavaScript. Posiada wiedzę niezbędną do rozumienia struktury i zasad działania aplikacji skryptowych w języku Python, Perl i JavaScript.	IST_O1_K_W05	P6S_WG P6S_WG_inż
W2	Ma uporządkowaną i podbudowaną teoretycznie wiedzę w zakresie jakości i cyklu życia aplikacji w językach Python, Perl i JavaScript.	IST_O1_K_W12	P6S_WG P6S_WG_inż
W3	Orientuje się w obecnym stanie i najnowszych trendach rozwojowych informatyki, rozumie model pracy z językami skryptowymi ich słabości i mocne strony.	IST_O1_K_W16	P6S_WG P6S_WG_inż
Umiejętności:			
U1	Potrafi samodzielnie nauczyć się wybranych elementów i struktur nowego języka programowania. Potrafi sprawnie wyszukać i wykorzystać informacje niezbędne do poznania nowego zagadnienia lub rozwiązania problemu.	IST_O1_K_U05, IST_O1_K_U14	P6S_UW, P6S_UW_inż, P6S_UW P6S_UW_inż
U2	Potrafi sformułować wymagania dla programów pisanych w językach skryptowych, opracować model oprogramowania oraz cenić prosty system informatyczny ze względu na ich wydajność, modularność i skalowalność.	IST_O1_K_U04, IST_O1_K_U12	P6S_UW, P6S_UW_inż, P6S_UW P6S_UW_inż
U3	Potrafi dostosować podejście programistyczne do rozwiązywanego problemu, potrafi ocenić w jakich sytuacjach języki skryptowe nie powinny być stosowane, potrafi uzasadnić użycie języków skryptowych ze względu na ich wydajność, modularność i skalowalność	IST_O1_K_U20	P6S_UK
Kompetencje społeczne:			
K1	Rozumie potrzebę i zna możliwości sprawnego wyszukiwania i wykorzystania informacji niezbędnych do poznania nowego zagadnienia lub rozwiązania problemu.	IST_O1_K_K01	P6S_KK
K2	Potrafi współdziałać i pracować w grupie, przyjmując w niej różne role.	IST_O1_K_K05	P6S_KR

3. Treści programowe

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
1.	- Charakterystyka języków skryptowych, działanie interpretera. - Składnia języka Python, podstawowe typy danych, listy i krotki, operatory logiczne, operatory arytmetyczne, instrukcje warunkowe i pętle, identyfikatory, liczby całkowite, operatory bitowe, typ float, complex i decimal, ciągi tekstowe, funkcje, moduły, klasy i obiekty, komunikacja przez internet, wyrażenia regularne, wątki i procesy, współpraca z bazami danych, interfejs graficzny. - Składnia języka Perl, podstawowe typy, pętle, operacje na plikach, operacje na ciągach znaków, funkcje, zasięg zmiennych. - JavaScript jako język dynamiczny, typy danych, deklaracje zmiennych, literały różnych typów, mechanizm automatycznej konwersji typów danych, obliczanie wartości wyrażeń logicznych (short circuit evaluation), tablice – indeksowanie, rozszerzanie, kopiowanie dla typów prostych i obiektowych, podstawowe bloki organizacji kodu (if, for, while, case), funkcje – parametry, operator reszty, wywołania zwrotne, funkcje natychmiastowe, domknięcia (closures), funkcje strzałkowe, zakresy zmiennych, wynoszenie zmiennych (hoisting), obiekty – konstruktory, dodawanie/usuwanie właściwości, dostęp do właściwości. Składnia Języka JavaScript,	Wykład	W1, W2, W3, K1, K2
2.	Ćwiczenia laboratoryjne: - Python: Tworzenie nowego programu, komentarze, typy danych, słowa kluczowe; - Python: Podstawowe działania na listach, krotkach i zbiorach; - Python: Definiowanie klas, metody klas, dziedziczenie; - Python: Wyrażenia regularne; - Python: Bazy danych, połączenia z bazą, operacje na danych; - Python: Wątki, tworzenie, synchronizacja; - Perl: typy zmiennych, formatowania, operatory; - Perl: Pętle, strumienie, operacje na stringach; - Funkcje w Perlu; - JS: zmienne, typy danych, operatory, obiekty; - JS: DOM, jQuery	Ćwiczenia laboratoryjne	U1, U2, U3, K1, K2

4. Metody prowadzenia zajęć, weryfikacji efektów uczenia się i warunki zaliczenia

Forma zajęć		
Wykład	Metody prowadzenia zajęć:	
	Wykład, Dyskusja, Pokaz	
	Metody (sposoby) weryfikacji:	Udział:
	Zaliczenie pisemne	100%
	Warunki zaliczenia przedmiotu:	
	Wykład zaliczany na podstawie dwóch kolokwίων pisemnego. Warunkiem zaliczenia jest uzyskanie min. 51% punktów z każdego kolokwium. Ocena końcowa to średnia arytmetyczna z dwóch kolokwίων.	

Ćwiczenia laboratoryjne	Metody prowadzenia zajęć:	
	Dyskusja, Ćwiczenia laboratoryjne	
	Metody (sposoby) weryfikacji:	Udział:
	Sprawozdanie	100%
	Warunki zaliczenia przedmiotu:	
	Laboratorium zaliczane na podstawie sprawozdań. Warunkiem zaliczenia jest oddanie wszystkich sprawozdań. Ocena końcowa to średnia arytmetyczna z ocen cząstkowych za sprawozdania.	

Efekt uczenia się dla przedmiotu	Metody (sposoby) weryfikacji	
	Zaliczenie pisemne	Sprawozdanie
W1	x	
W2	x	
W3	x	
U1		x
U2		x
U3		x
K1	x	x
K2	x	x

5. Literatura

Literatura podstawowa

1. Gorelick M., Ozsvald I., 2015, Python : programuj szybko i wydajnie, Helion, Gliwice
2. Cozens S., 2006, Perl : zaawansowane programowanie, Helion, Gliwice:
3. Chinnathambi K., 2017, JavaScript : przewodnik dla absolutnie początkujących, Helion, Gliwice

Literatura uzupełniająca

1. Lutz M., 2014, Python : leksykon kieszonkowy, Helion, Gliwice
2. Langworth I. & chromatic, 2006, Perl : testowanie, HELION, Gliwice
3. Zakas N.C., 2014, JavaScript : zasady programowania obiektowego, HELION, Gliwice

6. Nakład pracy studenta - bilans godzin i punktów ECTS

Aktywność studenta	Obciążenie studenta Liczba godzin
--------------------	--------------------------------------

Zajęcia prowadzone z bezpośrednim udziałem nauczyciela akademickiego lub innych osób prowadzących zajęcia	Wykład	30
	Ćwiczenia laboratoryjne	30
Praca własna studenta	Konsultacje	4
	Przygotowanie do zajęć	26
	Studiowanie literatury	8
	Przygotowanie do zaliczenia	12
	Przygotowanie sprawozdania	15
Łączny nakład pracy studenta		125
Liczba punktów ECTS		5

* Godzina (dydaktyczna) oznacza 45 minut