

KARTA KURSU

Fizyka

Studia I stopnia
2020/2021

Nazwa	Podstawy programowania 2
Nazwa w j. ang.	<i>Introduction to programming 2</i>

Koordynator	dr Grzegorz Stachowski	Zespół dydaktyczny
		dr Dawid Naęcz dr Grzegorz Stachowski
Punktacja ECTS*	2	

Opis kursu (cele kształcenia)

Celem przedmiotu jest zapoznanie słuchaczy z językiem Python i jego zastosowaniami do obliczeń naukowych oraz symulacji wykonywanych na użytek fizyki i techniki.
Po zakończeniu kursu student powinien znać podstawy programowania w języku Python i umieć je stosować.
Przedmiot prowadzony jest w języku polskim.

Warunki wstępne

Wiedza	Student zna podstawy programowania strukturalnego.
Umiejętności	Student posiada umiejętność napisania prostych programów w dowolnym języku proceduralnym z zastosowaniem podstawowych struktur logicznych oraz umiejętność opisanie prostych algorytmów.
Kursy	Podstawy programowania 1

Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	Po zakończeniu kursu student:	
	W 01: zna podstawy programowania, w tym, rozumie ideę programowania strukturalnego, proceduralnego i obiektowego w nowoczesnym języku programowania wysokiego poziomu (Python).	K_W05, K_W06, K_W10
	W 02: wie na czym polega działanie interpretera języka, jak uruchamiać programy w Pythonie i integrować je z resztą systemu.	K_W05, K_W06
	W 03: zna wybrane moduły języka Python rozszerzające jego działanie.	KW_05, KW_06
	W 04: student zna konstrukcje struktur danych takich jak lista, tuple, zbiór, słownik i umie je odpowiednio wykorzystać.	K_W05, K_W06

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	Po zakończeniu kursu student:	
	U01: potrafi przetłumaczyć projekt algorytmu w działający program w języku Python i wykonać proste symulacje.	K_U05, K_U07, K_U09,
	U02: samodzielnie pisze proste programy w języku Python (z zachowaniem stylu gwarantującego czytelność kodu), w których: -poprawnie definiuje i wykorzystuje struktury danych takie jak: tablice jednowymiarowe i wielowymiarowe, listy, zbiory, słowniki. -poprawnie dołącza moduły, definiuje i wywołuje funkcje.	K_U05, K_U07, K_U09,
	U03: umie instalować dodatkowe moduły i uruchamiać programy w języku Python, oraz poprawiać błędy składniowe oraz błędy wykonania.	K_U05, K_U07, K_U09

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Kompetencje społeczne	Po zakończeniu kursu student:	
	K01: potrafi korzystać z różnych źródeł informacji (w tym zasobów sieci Internet) do poszerzania własnej wiedzy i zdobywania nowych umiejętności z zakresu programowania.	K_K02
	K02: ma zdolność kreatywnego myślenia i rozwiązywania problemów programistycznych zarówno w pracy indywidualnej jak i zespołowej.	K_K04
	K03: potrafi przekazywać wiedzę informatyczną w sposób zrozumiały dla innych, potrafi wskazać związki pomiędzy fizyką oraz informatyką oraz ich wpływ na rozwój ludzkości.	K_K07

Organizacja										
Forma zajęć	Wykład (W)	Ćwiczenia w grupach								
		A		K		L		S		P
Liczba godzin						30				

Opis metod prowadzenia zajęć

Podczas zajęć laboratoryjnych studenci rozwiązują problemy zdefiniowane przez prowadzącego zajęcia.

Ćwiczenia nieukończone w trakcie zajęć studenci kończą samodzielnie (zdalnie na dedykowanym serwerze) oraz przesyłają na wskazany adres e-mail przed kolejnymi zajęciami.

Właściwa praca laboratoryjna poprzedzona jest przedstawieniem potrzebnych zagadnień i przykładów.

Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne
W01					X	X							
W02					X	X							
W03					X	X							
W04					X	X							
W05					X	X							
U01					X	X							
U02					X	X							
U03					X	X							
K01					X	X							
K02					X	X							
K03					X	X							

Kryteria oceny	Ocenę dobrą lub bardzo dobrą może uzyskać student, który wykaże się dobrą lub bardzo dobrą znajomością języka Python oraz umiejętnością samodzielnego konstruowania programów. Ocena końcowa jest średnią ocen za poszczególne zadania wykonywane w trakcie ćwiczeń laboratoryjnych.
----------------	---

Uwagi	
-------	--

Treści merytoryczne (wykaz tematów)

<p>Środowisko programowania języka Python Rodzaje i struktury danych w języku Python Polecenia w Pythonie Zbiory danych. Funkcje i struktury kontrolne. Moduły. Wstęp do programowania obiektowego. Obsługa plików. Pakiety naukowe w Pythonie (numpy, Matplotlib). Opracowanie danych pomiarowych w Pythonie: statystyka, wykresy, dopasowanie funkcji Proste symulacje wybranych procesów fizycznych</p>
--

Wykaz literatury podstawowej

<p>M.Summerfield, Python 3, Kompletne wprowadzenie do programowania, Helion, M.Lutz, D.Asher, Python. Wprowadzenie, Helion M.Lutz, Python. Leksykon kieszonkowy, Helion</p>

Wykaz literatury uzupełniającej

<p>Marek Gągolewski, Maciej Bartoszuł, Anna Cena, Przetwarzanie i analiza danych w języku Python, PWN K.J.Millman and M.Aivazis, "Python for scientists and engineers".</p>
--

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta)

liczba godzin w kontakcie z prowadzącymi	Wykład	0
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	10
liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	10
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	0
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	0
	Przygotowanie do egzaminu/zaliczenia	10
Ogółem bilans czasu pracy		60
Liczba punktów ECTS w zależności od przyjętego przelicznika 1ECTS=30h		2