

Karta przedmiotu

| | | | | | | | |
|--|---|---|------------------------|--------------|--|------------|-------|
| Nazwa i kod przedmiotu | Programowanie , PG_00182290 | | | | | | |
| Kierunek studiów | Fizyka (O) | | | | | | |
| Data rozpoczęcia studiów | październik 2026 r. | Rok akademicki realizacji przedmiotu | | | 2026/2027 | | |
| Poziom kształcenia | I stopnia - licencjackie | Grupa zajęć | | | Grupa zajęć obowiązkowych z zakresu kierunku studiów | | |
| Forma studiów | stacjonarne | Sposób realizacji | | | na uczelni | | |
| Rok studiów | 1 | Język wykładowy | | | polski | | |
| Semestr studiów | 1 | Liczba punktów ECTS | | | 3.0 | | |
| Profil kształcenia | ogólnoakademicki | Forma zaliczenia | | | zaliczenie | | |
| Jednostka prowadząca | Rektor -> Wydział Matematyki, Fizyki i Informatyki -> Instytut Fizyki Doświadczalnej -> Zakład Fizykochem.Pow. i Zjaw. Międzyfaz. | | | | | | |
| Imię i nazwisko wykładowcy (wykładowców) | Odpowiedzialny za przedmiot | dr Adrian Kołodziejski | | | | | |
| | Prowadzący zajęcia z przedmiotu | | | | | | |
| Formy zajęć | Forma zajęć | Wykład | Ćwiczenia | Laboratorium | Projekt | Seminarium | RAZEM |
| | Liczba godzin zajęć | 0.0 | 0.0 | 45.0 | 0.0 | 0.0 | 45 |
| | W tym liczba godzin zajęć na odległość: 0.0 | | | | | | |
| Aktywność studenta i liczba godzin pracy | Aktywność studenta | Udział w zajęciach dydaktycznych, objętych planem studiów | Udział w konsultacjach | | Praca własna studenta | | RAZEM |
| | Liczba godzin pracy studenta | 45 | 0.0 | | 30.0 | | 75 |
| Cel przedmiotu | Celem zajęć jest nabycie wiedzy i umiejętności z zakresu programowania z użyciem wybranego języka wysokiego poziomu. | | | | | | |

| Efekty uczenia się przedmiotu | Efekt kierunkowy | Efekt z przedmiotu | Sposób weryfikacji i oceny efektu |
|-------------------------------|---|---|--|
| | [FIZL3_K02] potrafi precyzyjnie formułować problemy służące pogłębieniu zrozumienia danego tematu | Student potrafi: - precyzyjnie sformułować problem obliczeniowy do dalszej analizy; - zaimplementować sformułowany problem w wybranym języku programowania; - potrafi analizować i interpretować wyniki w celu pogłębienia zrozumienia badanego zagadnienia; | [SK1] wypowiedź ustna/rozmowa/diskusja [SK2] prezentacja/projekt/referat/raport [SK5] realizacja zadania problemowego [SK6] demonstracja umiejętności praktycznych [SK8] obserwacja samodzielnej lub zespołowej pracy studenta |
| | [FIZL3_W12] zna metody analizy numerycznej, zna na poziomie zaawansowanym co najmniej jeden pakiet do obliczeń symbolicznych, zna pakiety oprogramowania użytkowego do prezentacji wyników i analizy danych; zna w sposób biegły jeden język programowania | Student: - ma wiedzę na temat programowania strukturalnego i obiektowego w wybranym języku wysokiego poziomu (Python); - zna typy danych, instrukcje sterujące oraz operatory arytmetyczne i logiczne w języku Python; - rozumie zagadnienia programowania obiektowego oraz potrafi je stosować w praktycznych zadaniach; - zna i potrafi wykorzystać pakiety obliczeniowe oraz narzędzia do obliczeń symbolicznych; - zna i potrafi korzystać z pakietów oprogramowania użytkowego do prezentacji wyników i analizy danych. | [SW1] wypowiedź ustna/rozmowa/diskusja [SW2] prezentacja/projekt/referat/raport [SW5] realizacja zadania problemowego |
| | [FIZL3_U12] potrafi skompilować, uruchomić, testować i udokumentować napisany samodzielnie program komputerowy | Student potrafi: - sformułować prosty algorytm numeryczny rozwiązujący zadany problem; - napisać i uruchomić w wybranym środowisku programistycznym program komputerowy w języku Python, przetwarzający dane liczbowe i tekstowe; - stosować podejście strukturalne i obiektowe w tworzeniu programu gotowego do testowania i dokumentowania; - napisać program odczytujący/zapisujący dane alfanumeryczne do/z pliku; - przetestować program, zidentyfikować i poprawić błędy oraz sporządzić dokumentację opisującą sposób działania programu. | [SU1] wypowiedź ustna/rozmowa/diskusja [SU2] prezentacja/projekt/referat/raport [SU5] realizacja zadania problemowego [SU6] demonstracja umiejętności praktycznych [SU8] obserwacja samodzielnej lub zespołowej pracy studenta |
| Treści przedmiotu | <ol style="list-style-type: none"> 1. Wstęp do Pythona i środowiska Spyder. 2. Typy danych, operatory arytmetyczne i logiczne, instrukcje warunkowe i sterujące. 3. Złożone typy danych: listy, krotki, słowniki. 4. Programowanie proceduralne: funkcje, moduły, podstawowy pracy z pakietami. 5. Odczyt i zapis danych alfanumerycznych do i z pliku. 6. Elementy programowania obiektowego: klasy, obiekty, metody. 7. Podstawy pracy z pakietami obliczeniowymi i wizualizacji danych. | | |
| Wymagania wstępne i dodatkowe | Podstawowa znajomość matematyki i fizyki, umiejętność obsługi komputera oraz logicznego myślenia, gotowość do samodzielnej nauki i rozwijania umiejętności programistycznych. | | |

| Sposoby i kryteria oceniania osiągniętych efektów uczenia się | Sposób oceniania (składowe) | Próg zaliczeniowy | Składowa oceny końcowej |
|---|--|---|-------------------------|
| | zadania domowe | 0.0% | 20.0% |
| | wykonywanie zadań programistycznych w trakcie semestru | 51.0% | 80.0% |
| Zalecana lista lektur | Podstawowa lista lektur | <p>A.1. Wykorzystywana podczas zajęć:</p> <ul style="list-style-type: none"> Instrukcje i materiały udostępniane przez prowadzącego. <p>A.2. Studiowana samodzielnie przez studenta:</p> <ul style="list-style-type: none"> Lutz, M., <i>Python. Wprowadzenie</i>, Helion, 2022. Matthes, E., <i>Python. Instrukcje dla programisty</i>, Helion, 2019. Dziedzic, K., <i>Python w pigułce</i>, Ringier Axel Springer, 2021. Kubiak, M., <i>Python. Zadania z programowania</i>, Helion, 2021. Dokumentacja Pythona: https://docs.python.org/3/ Materiały online dotyczące pakietów obliczeniowych i wizualizacji danych, np. NumPy i Matplotlib. | |
| | Uzupełniająca lista lektur | R. Johansson, <i>Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib</i> . Helion 2019 | |
| | Adresy eZasobów | | |
| Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania | <p>Napisz program w języku Python, który:</p> <ol style="list-style-type: none"> Wczyta dane pomiarowe z pliku CSV (np. czas, temperatura, napięcie). Przeprowadzi prostą analizę: obliczy średnie wartości, odchylenia, minimalne i maksymalne wartości dla każdej kolumny. Utworzy wykresy zależności między danymi (np. temperatura vs. napięcie). | | |
| Praktyki zawodowe w ramach przedmiotu | Nie dotyczy | | |

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.