Uniwersytet Gdański

Karta przedmiotu

Nazwa i kod przedmiotu	Wprowadzenie do programowania (Ćw. audytoryjne), PG_00188606						
Kierunek studiów	Filologia angielska (O)						
Data rozpoczęcia studiów	październik 2026 r.		Rok akademicki realizacji przedmiotu	2026/2027			
Poziom kształcenia	II stopnia		Grupa zajęć	Grupa zajęć obowiązkowych z zakresu kierunku studiów Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki			
Forma studiów	stacjonarne		Sposób realizacji	na uczelni			
Rok studiów	1		Język wykładowy	polski			
Semestr studiów	2		Liczba punktów ECTS	4.0			
Profil kształcenia	ogólnoakademicki		Forma zaliczenia	zaliczenie			
Jednostka prowadząca	Rektor -> Wydział Filologiczny -> Instytut Anglistyki i Amerykanistyki -> Zakład Językoznawstwa Korpusowego i Glottodydaktyki						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		mgr Aleksandra Chmarzyńska				
	Prowadzący zajęcia z przedmiotu						
Formy zajęć	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	0.0	60.0	0.0	0.0	0.0	60
	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	60		4.0		36.0	100
Cel przedmiotu	Celem przedmiotu jest zaznajomienie studentow z podstawowymi technikami i metodami programowania, na przykładzie jezyka programowania Python.						

Efekty uczenia się przedmiotu

Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
[FAMU2_W12] Zna i rozumie wybrane zagadnienia z zakresu nauk pomocniczych i pokrewnych filologii angielskiej oraz realioznawstwa krajów anglojęzycznych, niezbędne jako kontekst dla badań w zakresie anglistycznego językoznawstwa i literatur anglojęzycznych lub w działalności zawodowej filologa anglisty.	Zna i rozumie wybrane zagadnienia z zakresu programowania.	[SW5] realizacja zadania problemowego
[FAMU2_K02] Jest gotowa do uznawania znaczenia wiedzy i umiejętności z zakresu studiów anglistycznych w rozwiązywaniu problemów poznawczych i praktycznych oraz do zasięgania opinii opiekuna naukowego lub opiekuna w wybranym miejscu pracy w przypadku trudności z samodzielnym rozwiązywaniem problemów.	Jest gotów do uznawania znaczenia wiedzy i umiejętności z zakresu programowania w rozwiązywaniu problemów poznawczych i praktycznych oraz do zasięgania opinii nauczyciela akademickiego w przypadku trudności z samodzielnym rozwiązywaniem problemów.	[SK5] realizacja zadania problemowego
[FAMU2_K01] Jest gotowa do krytycznej oceny zakresu posiadanej przez siebie wiedzy i umiejętności, w szczególności z zakresu anglojęzycznego językoznawstwa i literaturoznawstwa oraz języka angielskiego.	Jest gotów do krytycznej oceny zakresu posiadanej przez siebie wiedzy i umiejętności z zakresu programowania.	[SK5] realizacja zadania problemowego
[FAMU2_U10] Potrafi samodzielnie planować i realizować własne uczenie się przez całe życie i ukierunkowywać innych w tym zakresie w ramach filologii angielskiej i wybranej sfery działalności zawodowej.	Potrafi samodzielnie planować i realizować własne uczenie się przez całe życie i ukierunkowywać innych w zakresie programowania	[SU5] realizacja zadania problemowego
[FAMU2_U04] Potrafi dobierać i stosować techniki informacyjno-komunikacyjne (ICT) podczas pozyskiwania i przetwarzania informacji dla celów badawczych i zawodowych w ramach filologii angielskiej.	Potrafi dobierać i stosować techniki informacyjno-komunikacyjne (ICT) w zakresie programowania.	[SU5] realizacja zadania problemowego

Treści przedmiotu

- Pojecie programu. Instalacja i wprowadzenie do srodowiska jezyka Python.
- Podstawowe typy zmiennych.
- Pojecie funkcji w programowaniu.
- Sterowanie programem za pomoca instrukcji warunkowych w programowaniu.
- Struktury cykliczne w jezyku Python: petle for oraz while.
- Zaawansowane struktury danych: listy, krotki, zbiory oraz słowniki.
- Operacje wejscia-wyjscia oraz na plikach w jezyku Python.
- Pojecie algorytmu i schematu blokowego. Podstawowe algorytmy wyszukiwania oraz sortowania.
- Technika dziel i zwyciezaj w programowaniu.
- Podstawowe algorytmy tekstowe.
- Pojecie rekurencji w programowaniu.
- Zarzadzanie bibliotekami w jezyku Python.
- Wyrazenia regularne w jezyku Python.
- Obsługa plikow tekstowych w jezyku Python.
- Format JSON oraz ich uzycie w jezyku Python
- Zastosowanie bibliotek: numpy, nltk, matplotlib, seaborn, pygal, spacy

Wymagania wstępne i dodatkowe

OBECNOŚĆ NA WSZYSTKICH ZAJĘCIACH JEST OBOWIĄZKOWA, O ILE OSOBA STUDIUJĄCA NIE UZYSKA ZGODY DZIEKANA NA STUDIOWANIE WEDŁUG INDYWIDUALNEJ ORGANIZACJI STUDIÓW.

Sposoby i kryteria oceniania osiąganych efektów uczenia się

Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
Praca zaliczeniowa (zadania)	51.0%	100.0%

<table>
<tr><td rowspan="3">Zalecana lista lektur</td><td>Podstawowa lista lektur</td><td>P. Broda, D. Smołucha, Informatyka. Podrecznik czesc II, Wyd. Operon, 2006.

M. Dawson, Python dla kazdego. Podstawy programowania, Wyd. Helion, 2010.

T. Gaddins, Python dla zupełnie poczatkujacych, Wyd. Helion, 2019.

E. Grubiel, G. Hardt-Olejniczak, E. Kołczyk, H. Krupicka, M. M. Sysło, Informatyka czesc I, Wyd. WSiP, 2002.

A. Sweigart, Automatyzacja nudnych zadan z Pythonem. Nauka programowania, Wyd. Helion, 2015.

A. Szepietowski, Matematyka dyskretna, Wyd. UG 2004.

P. Wroblewski, Algorytmy, struktury danych i techniki programowania, Wyd. Helion, 2010.</td></tr>
<tr><td>Uzupełniająca lista lektur</td><td>T. H. Cormen, Ch. E. Leiserson, R. L. Rivest, Wprowadzenie do algorytmow, Wyd. Naukowe PWN, 2012.

P. Krugiołka, Linux. Jak dostroic bestie do swoich potrzeb, Wyd. Helion, 2012.

E. Matthes, Python. Instrukcje dla programisty, Wyd. Helion, 2016.</td></tr>
<tr><td>Adresy eZasobów</td><td></td></tr>
<tr><td>Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania</td><td colspan="2">1. Pojęcie programu. Instalacja i wprowadzenie do środowiska języka Python:
• Co to jest program komputerowy?
• Jakie są kroki instalacji środowiska Python
• Wprowadzenie do IDE (np. PyCharm, VS Code, Jupyter Notebook).
1. Podstawowe typy zmiennych:
• Jakie są podstawowe typy danych w Pythonie (int, float, str, bool)?
• Jakie operacje można wykonywać na różnych typach danych?
• Ćwiczenie: Napisz program, który przyjmuje liczby całkowite od użytkownika i wykonuje podstawowe operacje arytmetyczne.
1. Pojęcie funkcji w programowaniu:
• Jak definiować funkcje w Pythonie?
• Co to są argumenty i wartości zwracane przez funkcję?
• Ćwiczenie: Napisz funkcję, która oblicza silnię liczby.
1. Sterowanie programem za pomocą instrukcji warunkowych w programowaniu:
• Jak działają instrukcje warunkowe if, elif, else?
• Ćwiczenie: Napisz program, który sprawdza, czy liczba jest parzysta czy nieparzysta.
1. Struktury cykliczne w języku Python: pętle for oraz while:
• Jak działa pętla for i while?
• Ćwiczenie: Napisz program, który wyświetla liczby od 1 do 10 przy użyciu obu typów pętli.
1. Zaawansowane struktury danych: listy, krotki, zbiory oraz słowniki:
• Jakie są różnice między listami, krotkami, zbiorami a słownikami?
• Ćwiczenie: Napisz program, który sortuje listę liczb i usuwa duplikaty.
1. Operacje wejścia-wyjścia oraz na plikach w języku Python:
• Jak otwierać, czytać i zapisywać pliki tekstowe w Pythonie?
• Ćwiczenie: Napisz program, który zapisuje dane użytkownika do pliku tekstowego.</td></tr>
<tr><td>Praktyki zawodowe w ramach przedmiotu</td><td colspan="2">Nie dotyczy</td></tr>
</table>

Dokument wygenerowany elektronicznie. Nie wymaga pieczęci ani podpisu.