Opis **zajęć (sylabus)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Nazwa zajęć: | | Programowanie skryptowe | | | | | | | **ECTS** | **3** |
| Nazwa zajęć w j. angielskim: | | Scripting programming | | | | | | | | |
| Zajęcia dla kierunku studiów: | | **Informatyka i Ekonometria** | | | | | | | | |
|  | |  | | | | | | | | |
| Język wykładowy: | | polski | | | | Poziom studiów: | | studia I stopnia | | |
| Forma studiów: | stacjonarne  niestacjonarne | Status zajęć: |  podstawowe   kierunkowe | obowiązkowe  do wyboru | | Numer semestru: ……**4**….. | |  semestr zimowy  semestr letni | | |
|  |  | Rok akademicki, od którego obowiązuje opis (rocznik): | | | | 2019/2020 | Numer katalogowy: | **ZIM-IE-1Z-04L-28** | | |
|  | | | | | | | | | | |
| Koordynator zajęć: | |  | | | | | | | | |
| Prowadzący zajęcia: | |  | | | | | | | | |
| Jednostka realizująca: | |  | | | | | | | | |
| Jednostka zlecająca: | |  | | | | | | | | |
| Założenia, cele i opis zajęć: | | Celem przedmiotu jest zapoznanie studentów z podstawowymi aspektami programowania w językach skryptowych. Podczas zajęć planowany jest przegląd podstawowych konstrukcji programistycznych oraz struktur danych wykorzystywanych w wybranym skryptowym języku programowania w ujęciu proceduralnym oraz obiektowym.Tematyka zajęć:   1. Języki skryptowe. Interpreter języka programowania. Tryb interaktywny oraz tryb skryptowy. 2. Podstawy programowania proceduralnego w wybranym języku skryptowym: wprowadzanie i wyprowadzanie danych, instrukcja warunkowa, operacje arytmetyczno – logiczne. Wykorzystanie funkcji matematycznych.3. Instrukcja iteracyjna WHILE.4. Instrukcje iteracyjne FOR (licznikowa oraz zakresowa).5. Struktury sekwencyjne - krotki, łańcuchy znaków, listy oraz słowniki.6. Funkcje i ich wykorzystanie. Tworzenie własnych modułów i pakietów.7. Programy z argumentami pozycyjnymi.8. Obsługa wyjątków.9. Operacje na plikach.10. Podstawy programowania obiektowego: klasy, metody, obiekty i pola, konstruktory i atrybuty klas, polimorfizm i dziedziczenie.11. Tworzenie interfejsów graficznych GUI.12. Wykorzystanie elementów graficznych i multimedialnych w językach skryptowych. | | | | | | | | |
| Formy dydaktyczne, liczba godzin: | | ćwiczenia laboratoryjne; liczba godzin ...18... | | | | | | | | |
| Metody dydaktyczne: | | dyskusja problemu, rozwiązywanie problemu, studium przypadku, projekt indywidualny, konsultacje | | | | | | | | |
| Wymagania formalne  i założenia wstępne: | | Wymagana jest wiedza z zakresu przedmiotów: Wstęp do programowania, Programowanie obiektowe, Algorytmy i struktury danych | | | | | | | | |
| Efekty uczenia się: | | Wiedza:  1 – posiada podstawową wiedzę dotyczącą struktur programistycznych oraz struktur danych wykorzystywanych w skryptowych językach programowania  2 – wie, w jaki sposób napisać skrypt, wykorzystując proceduralny oraz obiektowy paradygmat programowania  3 – zna sposoby tworzenia graficznych interfejsów użytkownika (GUI) w wybranym skryptowym języku programowania | | | Umiejętności:  1 – potrafi formułować rozwiązanie danego problemu matematycznego w postaci algorytmu wykorzystującego podstawowe konstrukcje programistyczne oraz struktury danych  2 - potrafi napisać skrypt rozwiązujący dane zagadnienie matematyczne z wykorzystaniem wybranego skryptowego języka programowania  3 – potrafi zaprojektować prostą aplikację z interfejsem graficznym (GUI) | | | | Kompetencje:  1 – podejmuje samodzielną decyzję wyboru odpowiedniego języka skryptowego jako narzędzia wykorzystywanego do rozwiązania danego problemu | |
| Sposób weryfikacji efektów uczenia się: | | Pisemny test wiedzy ze znajomości specyfiki programowania proceduralnego oraz obiektowego w wybranym skryptowym języku programowania (efekty wiedzy)  Kolokwium praktyczne polegające na napisaniu określonej liczby skryptów rozwiązujących postawione problemy (efekty umiejętności 1 oraz 2)  Indywidualny projekt aplikacji wykorzystującej interfejs graficzny (GUI) (efekt umiejętności 3) | | | | | | | | |
| Forma dokumentacji osiągniętych efektów uczenia się: | | Ocenione prace studentów z pisemnego testu wiedzy  Ocenione pliki kolokwium praktycznego w wersji elektronicznej  Ocenione i zarchiwizowane pliki indywidualnych projektów studenckich | | | | | | | | |
| Elementy i wagi mające wpływ  na ocenę końcową: | | **Kolokwium praktyczne – 50%**  **Projekt indywidualny – 25%**  **Test wiedzy – 25%** | | | | | | | | |
| Miejsce realizacji zajęć: | | laboratorium komputerowe | | | | | | | | |
| Literatura podstawowa i uzupełniająca:  Literatura podstawowa:  - Michael Dawson: Python dla każdego. Podstawy programowania. Od zera do bohatera, Helion 2014.  - Dive into Python – internetowy podręcznik programowania w Pythonie  Literatura uzupełniająca:  - materiały prezentacyjne w postaci plików udostępniane przez prowadzącego | | | | | | | | | | |
| UWAGI  Minimalna liczba punktów konieczna do zaliczenia: 50% z kolokwium praktycznego, 50% z projektu indywidualnego oraz 50% z testu wiedzy | | | | | | | | | | |

Wskaźniki ilościowe charakteryzujące moduł/przedmiot:

|  |  |
| --- | --- |
| Szacunkowa sumaryczna liczba godzin pracy studenta (kontaktowych i pracy własnej) niezbędna dla osiągnięcia zakładanych dla zajęć efektów uczenia się - na tej podstawie należy wypełnić pole ECTS: | **90 h** |
| Łączna liczba punktów ECTS, którą student uzyskuje na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich lub innych osób prowadzących zajęcia: | **1 ECTS** |

Tabela zgodności kierunkowych efektów uczenia się z efektami przedmiotu:

|  |  |  |  |
| --- | --- | --- | --- |
| kategoria efektu | Efekty uczenia się dla zajęć: | Odniesienie do efektów dla programu studiów dla kierunku | Oddziaływanie zajęć na efekt kierunkowy\*) |
| Wiedza 1 | posiada podstawową wiedzę dotyczącą struktur programistycznych oraz struktur danych wykorzystywanych w skryptowych językach programowania | K\_W12 / P6S\_WG | 2 |
| Wiedza 2 | wie, w jaki sposób napisać skrypt, wykorzystując proceduralny oraz obiektowy paradygmat programowania | K\_W13 / P6S\_WG | 3 |
| Wiedza 3 | zna sposoby tworzenia graficznych interfejsów użytkownika (GUI) w wybranym skryptowym języku programowania | K\_W13 / P6S\_WG | 3 |
| Umiejętności 1 | potrafi formułować rozwiązanie danego problemu matematycznego w postaci algorytmu wykorzystującego podstawowe konstrukcje programistyczne oraz struktury danych | K\_U13 / P6S\_UW | 3 |
| Umiejętności 2 | potrafi napisać skrypt rozwiązujący dane zagadnienie matematyczne z wykorzystaniem wybranego skryptowego języka programowania | K\_U15 / P6S\_UW | 2 |
| Umiejętności 3 | potrafi zaprojektować prostą aplikację z interfejsem graficznym (GUI) | K\_U16 / P6S\_UW | 3 |
| Kompetencje - 1 | podejmuje samodzielną decyzję wyboru odpowiedniego języka skryptowego jako narzędzia wykorzystywanego do rozwiązania danego problemu | K\_K06 / P6S\_KR | 1 |

\*)

3 – zaawansowany i szczegółowy,

2 – znaczący,

1 – podstawowy,