

KARTA KURSU

| | |
|-----------------|-------------------------|
| Nazwa | Języki skryptowe |
| Nazwa w j. ang. | Scripting languages |

| | | |
|-----------------|---|--|
| Koordynator | mgr Michał Frontczak | Zespół dydaktyczny |
| | | mgr Michał Frontczak mgr inż. Patryk Mieczkowski dr hab. inż. Mateusz Muchacki dr Wojciech Nawalaniec |
| Punktacja ECTS* | st. stacjonarne: 2 st. niestacjonarne: 2 | |

Opis kursu (cele kształcenia)

W trakcie zajęć studenci powinni się zapoznać z popularnym językiem skryptowe (na przykładzie języka Python). Kurs jest prowadzony w języku polskim.

Warunki wstępne

| | |
|--------------|--|
| Wiedza | Podstawowa wiedza o programowaniu strukturalnym. |
| Umiejętności | Umiejętność pisania prostych programów (np. w języku C). |
| Kursy | Podstawy programowania, Programowanie proceduralne. |

Efekty uczenia się

| | Efekt uczenia się dla kursu | Odniesienie do efektów kierunkowych |
|--------|--|-------------------------------------|
| Wiedza | Po zakończeniu kursu student: W01: zna składnię i działanie języka skryptowego Python; W02: rozumie zasadę działania i stosowania wyrażeń regularnych; W03: zna rodzaje testów i rozumie potrzebę ich stosowania; W04: zna popularne moduły języka Python. | K_W07 K_W07 K_W07 K_W07 |

| | Efekt uczenia się dla kursu | Odniesienie do efektów kierunkowych |
|--------------|--|-------------------------------------|
| Umiejętności | Po zakończeniu kursu student: | |
| | U01: potrafi utworzyć skrypty w języku Python, które umożliwiają automatyzację pewnych zadań; | K_U05 |
| | U02: potrafi wybrać odpowiednie konstrukcje języka Python pomocne w rozwiązaniu danego problemu; | K_U04, K_U05 |
| | U03: potrafi pisać testy z wykorzystaniem odpowiednich bibliotek języka Python; | K_U04, K_U05 |
| | U04: potrafi konstruować wyrażenia regularne oraz wykorzystać popularne moduły języka Python; | K_U04, K_U05 |

| | Efekt uczenia się dla kursu | Odniesienie do efektów kierunkowych |
|-----------------------|--|-------------------------------------|
| Kompetencje społeczne | Po zakończeniu kursu student: | |
| | K01: jest świadomy celowości i przydatności uzupełnienia wiedzy programistycznej. | K_K01 |
| | K02: jest świadomy korzyści płynących z umiejętności rozwiązywania problemów z zastosowaniem skryptów. | K_K01 |

Studia stacjonarne

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

Studia niestacjonarne

| | | Organizacja | | | | | | | | | | |
|---------------|------------|---------------------|--|---|--|----|--|---|--|---|--|---|
| Forma zajęć | Wykład (W) | Ćwiczenia w grupach | | | | | | | | | | |
| | | A | | K | | L | | S | | P | | E |
| Liczba godzin | | | | | | 20 | | | | | | |

Opis metod prowadzenia zajęć

Prowadzący przedstawia nowe zagadnienia oraz problemy wraz z ewentualnymi sugestiami dot. możliwych metod ich rozwiązania. Studenci piszą skrypt rozwiązujący zadany problem. Następnie odbywa się wspólna analiza rozwiązania.

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 | | X | | | | | |
| W02 | | | | | X | X | | X | | | | | |
| W03 | | | | | X | X | | X | | | | | |
| W04 | | | | | X | X | | X | | | | | |
| U01 | | | | | X | X | | X | | | | | |
| U02 | | | | | X | X | | X | | | | | |
| U03 | | | | | X | X | | X | | | | | |
| U04 | | | | | X | X | | X | | | | | |
| K01 | | | | | | | | X | | | | | |
| K02 | | | | | | | | X | | | | | |

| | |
|----------------|--|
| Kryteria oceny | <p>Podstawą do zaliczenia jest ocena uzyskana na podstawie oddanych zadań programistycznych lub projektu programistycznego (alternatywnie: na podstawie testu przygotowanego przez prowadzącego).</p> <p>Ocenę dobrą lub bardzo dobrą może uzyskać student, który otrzymał odpowiednią liczbę punktów z określonych zadań programistycznych (alternatywnie: uzyskał odpowiednią liczbę punktów na drodze testu przygotowanego przez prowadzącego).</p> |
|----------------|--|

| | |
|-------|--|
| Uwagi | |
|-------|--|

Treści merytoryczne (wykaz tematów)

| | |
|--|--|
| <ol style="list-style-type: none"> 1. 2. 3. 4. | <p>Instalacja języka Python i zarządzanie modułami. Środowisko PyCharm</p> <p>Język Python.</p> <ol style="list-style-type: none"> a. Wbudowane typy i struktury danych oraz ich różne odmiany. b. Sterowanie przebiegiem programu. c. Różne aspekty wykorzystania pętli. d. Funkcje i podstawowy mechanizm obsługi parametrów, słowa kluczowe. e. Rekurencja w języku Python f. Wyrażenia listowe i generatorowe. Filtrowanie i transformacja danych. g. Sortowanie danych i wykorzystanie funkcji anonimowych. h. Obsługa plików. Serializacja obiektów. i. Tworzenie modułów. j. Dekoratory funkcji i domknięcia, funkcje jako obiekty pierwszej klasy. k. Zakresy zmiennych (funkcja, moduł, domknięcie, itd.). l. Podstawowa obsługa wyjątków (konstrukcja try-except). m. Dopasowanie wzorców z użyciem match/case. <p>Wyrażenia regularne.</p> <ol style="list-style-type: none"> a. Obiekt dopasowania i jego metody. b. Grupy przechwytyjące. c. Dopasowania alternatywne, opcjonalne i wielokrotne. d. Zachłanne i niezachłanne dopasowanie. e. Klasy znaków. f. Znaki specjalne. g. Opcje dopasowania. <p>Testowanie i kontrola jakości</p> |
|--|--|

- a. Programowanie sterowane testami.
 - b. Testowanie interaktywnych przykładów dokumentacji – moduł doctest.
 - c. Pisanie testów z użyciem platformy pytest.
5. Wykorzystanie języka Python oraz jego modułów celem tworzenia skryptów, np:
- a. Automatyzacja zadań.
 - b. Manipulowanie plikami i katalogami.
 - c. Przetwarzanie plików graficznych.
 - d. Przetwarzanie plików tekstowych.

Wykaz literatury podstawowej

1. „Czysty kod w Pythonie. Twórz wydajny i łatwy w utrzymaniu kod. Wydanie II”, Mariano Anaya, Helion 2022 (wybrane rozdziały)
2. „Programowanie w Pythonie dla średnio zaawansowanych. Najlepsze praktyki tworzenia czystego kodu”, Al Sweigart, Helion 2022 (wybrane rozdziały)
3. „Kod Pythona w jednym wierszu. Jak profesjonaliści piszą programy doskonałe”, Ch. Mayer, Helion 2021 (wybrane rozdziały)
4. „Python. Instrukcje dla programisty. Wydanie II”, E. Matthes, Helion 2020 (wybrane fragmenty)
5. „Python. Wprowadzenie. Wydanie IV” M. Lutz, Helion 2010 (wybrane fragmenty)
6. „Python. Receptury”, D. Beazley, B. K. Jones, Helion 2014 (wybrane fragmenty)

Wykaz literatury uzupełniającej

1. „Python dla testera”, P. Wróblewski, Helion 2021
2. „Efektywny Python. 90 sposobów na lepszy kod. Wydanie II”, B. Slatkin, Helion 2020
3. „Czysty kod w Pythonie”, S. Kamil, Helion 2020
4. „Python, Dobre praktyki profesjonalistów”, D. Hillard, Helion 2020
5. „Python na poważnie”, J. Danjou, PWN 2019
6. „Python. Leksykon kieszonkowy. Wydanie V” M. Lutz, Helion 2014
7. „Python 3 : kompletne wprowadzenie do programowania”, M. Summerfield, Helion 2010

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) **studia stacjonarne**

| | | |
|---|--|----|
| liczba godzin w kontakcie z prowadzącymi | Wykład | |
| | Konwersatorium (ćwiczenia, laboratorium itd.) | 30 |
| | Pozostałe godziny kontaktu studenta z prowadzącym | 2 |
| liczba godzin pracy studenta bez kontaktu z prowadzącymi | Lektura w ramach przygotowania do zajęć | 8 |
| | Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu | |
| | Przygotowanie projektu lub prezentacji na podany temat (praca w grupie) | 10 |
| | Przygotowanie do egzaminu/zaliczenia | |
| Ogółem bilans czasu pracy | | 50 |
| Liczba punktów ECTS w zależności od przyjętego przelicznika | | 2 |

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) **studia niestacjonarne**

| | | |
|---|--|----|
| liczba godzin w kontakcie z prowadzącymi | Wykład | |
| | Konwersatorium (ćwiczenia, laboratorium itd.) | 20 |
| | Pozostałe godziny kontaktu studenta z prowadzącym | 5 |
| liczba godzin pracy studenta bez kontaktu z prowadzącymi | Lektura w ramach przygotowania do zajęć | 15 |
| | Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu | |
| | Przygotowanie projektu lub prezentacji na podany temat (praca w grupie) | 10 |
| | Przygotowanie do egzaminu/zaliczenia | |
| Ogółem bilans czasu pracy | | 50 |
| Liczba punktów ECTS w zależności od przyjętego przelicznika | | 2 |