**SYLABUS**

na cykl kształcenia rozpoczynający się w roku akademickim 2024/2025

|  |  |
| --- | --- |
| **Nazwa przedmiotu/modułu** | **Podstawy programowania I** |
| **Nazwa jednostki/-ek w której/ -ych jest przedmiot realizowany** | **Zakład Biostatystyki i Informatyki Medycznej** |
| **e-mail jednostki** | **statinfmed@umb.edu.pl** |
| **Wydział** | Lekarski z Oddziałem Stomatologii i Oddziałem Nauczania w Języku Angielskim |
| **Nazwa kierunku studiów** | Biostatystyka kliniczna |
| **Poziom kształcenia** | I stopnia |
| **Forma studiów** | stacjonarne ⬛ niestacjonarne □ |
| **Język przedmiotu** | polski ⬛ angielski □ |
| **Rodzaj przedmiotu** | obowiązkowy ⬛ fakultatywny □ |
| **Rok studiów/semestr** | I **⬛** II □ III □  | 1 □ 2 **⬛** 3 □ 4 □ 5 □ 6 □  |
| **Przedmioty wprowadzające wraz z wymaganiami wstępnymi** | Podstawy Logiki i Teorii Mnogości, Technologie Informacyjne, zaliczenie przedmiotów zgodnie z postępowaniem rekrutacyjnym |
| **Liczba godzin zajęć dydaktycznych z podziałem na formy prowadzenia zajęć** | Wykład: 15Ćwiczenia: 30 |
| **Założenia i cele przedmiotu** | Zapoznanie studentów z podstawami programowania |
| **Metody dydaktyczne** | Wykład: przekazywanie wiedzy w formie wykładu, prezentacja, samodzielne dochodzenie do wiedzy,Ćwiczenia: zajęcia praktyczne (ćwiczeniowe) przy stanowisku komputerowym, analiza literatury, prace domowe, projekt końcowy wraz z prezentacją i odpowiedzią ustną |
| **Imię i nazwisko osoby prowadzącej przedmiot** | Pracownicy naukowo-dydaktyczni i dydaktyczni zatrudnieni w Zakładzie Statystyki i Informatyki Medycznej |
| **Imię i nazwisko osoby odpowiedzialnej za dydaktykę** | dr hab. Robert Milewski  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **efekty uczenia się przedmiotowe (symbol** **i numer)** | **efekty uczenia się przedmiotowe****(opis)** | **Odniesienie do efektów kierunkowych (symbol i numer)**  | **Forma zajęć** | **Metody weryfikacji osiągnięcia zamierzonych efektów uczenia się** |
| **wiedza**  |
| W18 | Zna podstawowe konstrukcje programistyczne (wyrażenia, operacje, instrukcje). | K\_W17 | Wykład | Metody podsumowujące:- egzamin pisemny z części teoretycznej (pytania testowe, pytania otwarte),- egzamin przy stanowisku komputerowym w formie praktycznej - wykonanie projektu końcowego wraz z raportem i prezentacją.Metody formujące:- obserwacja pracy studenta,- ocena aktywności w czasie zajęć,- wykonywanie zadań powierzonych przez prowadzącego,- zaliczanie poszczególnych czynności,- prace domowe. |
| W19 | Zna podstawowe typy i struktury danych. | K\_W17 | Wykład |
| W20 | Zna podstawowe techniki programowania strukturalnego. | K\_W17 | Wykład |
| **umiejętności** |
| U17 | Potrafi stosować podstawowe konstrukcje programistyczne. | K\_U13 | Ćwiczenia | Metody podsumowujące:- egzamin pisemny z części teoretycznej (pytania testowe, pytania otwarte),- egzamin przy stanowisku komputerowym w formie praktycznej - wykonanie projektu końcowego wraz z raportem i prezentacją.Metody formujące:- obserwacja pracy studenta,- ocena aktywności w czasie zajęć,- wykonywanie zadań powierzonych przez prowadzącego,- zaliczanie poszczególnych czynności,- prace domowe. |
| U18 | Potrafi tworzyć proste programy strukturalne. | K\_U13 | Ćwiczenia |
| U19 | Potrafi korzystać z podstawowych typów i struktur danych. | K\_U13 | Ćwiczenia |
| **kompetencje społeczne** |
| K1 | Zna poziom własnych kompetencji i swoje ograniczenia w wykonywaniu zadań zawodowych oraz wie kiedy zasięgnąć opinii ekspertów. | K\_K01 | Wykład,Ćwiczenia | Metody podsumowujące:- ocenianie ciągłe przez nauczyciela,Metody formujące:- obserwacja pracy studenta,- bieżąca informacja zwrotna,- dyskusja |
| K2 | Efektywnie rozwiązuje postawione przed nim problemy, popierając je argumentacją w kontekście wybranych perspektyw teoretycznych oraz poglądów różnych autorów. | K\_K02 | Wykład,Ćwiczenia |
| K3 | Przestrzega zasady etyki zawodowej. | K\_K05 | Wykład,Ćwiczenia |

|  |  |
| --- | --- |
| **Punkty ECTS** | 4 |
| **Obciążenie pracą studenta** |
| **Forma aktywności** | **Liczba godzin na zrealizowanie aktywności**  |
| **Zajęcia wymagające udziału prowadzącego:** |
| 1. Realizacja przedmiotu: wykłady (wg planu studiów)
 | 15 |
| 1. Realizacja przedmiotu: ćwiczenia (wg planu studiów)
 | 30 |
| 1. Realizacja przedmiotu: seminaria (wg planu studiów)
 |  |
| 1. Realizacja przedmiotu: fakultety
 |  |
| 1. Udział w konsultacjach
 |  |
|  | godziny razem: 45 |
| **Samodzielna praca studenta:** |
| 1. Samodzielne przygotowanie się do zajęć teoretycznych i praktycznych (wykonanie projektu, dokumentacji, opisu przypadku itp.)
 | 35 |
| 1. Samodzielne przygotowanie się do zaliczeń/kolokwiów
 |  |
| 1. Samodzielne przygotowanie się do egzaminu/zaliczenia końcowego
 | 20 |
|  | godziny razem: 55 |

|  |
| --- |
| **Treści programowe przedmiotu:** |
| **efekty uczenia się przedmiotowe****(symbol i numer)** | **tematyka** |
| W18, K1, K2, K3 | Zagadnienia związane z tworzeniem oprogramowania: kod źródłowy, kompilacja, interpretacja. |
| W19, K1, K2, K3 | Zapoznanie się z podstawowymi strukturami danych stosowanych w danym języku programowania (typy proste, typy złożone), zapoznanie się z operatorami porównania, wyrażeniami, instrukcjami. |
| W20, K1, K2, K3 | Instrukcje warunkowe, iteracyjne oraz instrukcje wejścia – wyjścia. Polecenia i funkcje. |
| U17, K1, K2, K3 | Pisanie własnych programów, korzystanie z podstawowych konstrukcji programistycznych. |
| U18, K1, K2, K3 | Pisanie własnych programów z wykorzystaniem różnorodnych typów, operatorów, instrukcji, pętli i funkcji. |
| U19, K1, K2, K3 | Pisanie własnych programów, praca z podstawowymi typami i strukturami danych |

|  |
| --- |
| **Literatura podstawowa:** |
| 1. Eric Matthes, Python. Instrukcje dla programisty. Wydanie II, Helion, 2020
2. Paul Barry, Python. Rusz głową! Wydanie II, Helion, 2017
3. Mirosław J. Kubiak, Python. Zadania z programowania. Przykładowe funkcyjne rozwiązania, Helion, 2021
4. Mirosław J. Kubiak, Python. Zadania z programowania. Przykładowe imperatywne rozwiązania, Helion, 2021
 |
| **Literatura uzupełniająca:** |
| 1. Brett Slatkin, Efektywny Python. 90 sposobów na lepszy kod. Wydanie II, Helion, 2020
2. Dawson Michael, Python dla każdego. Podstawy programowania, Wydanie III, Helion, 2014
3. Mark Lutz, Python. Wprowadzenie. Wydanie V, Helion, 2020
 |

|  |
| --- |
| **Kryteria oceny osiągniętych efektów uczenia się oraz forma i warunki uzyskania zaliczenia przedmiotu:**  |
| Student zostaje dopuszczony do egzaminu z przedmiotu w momencie: kompletnego wykonania zadań powierzonych przez prowadzącego, zaliczeń prac domowych, wykonania projektu końcowego wraz z dostarczeniem i zaprezentowaniem raportu oraz odpracowania wszystkich nieobecności. Finalna ocena stanowi złożenie oceny uzyskanej z egzaminu oraz oceny uzyskanej z projektu końcowego. Wynik z egzaminu stanowi 60 % oceny końcowej, natomiast wynik z projektu końcowego stanowi 40 % oceny końcowej. Każda nieobecność musi zostać usprawiedliwiona i odpracowana w formie samodzielnego wykonania ćwiczeń w oparciu o literaturę przedmiotu lub konsultacje z osobą prowadzącą zajęcia.Szczegółowe warunki uzyskania oceny 3.0: Student uzyskał min. 50% sumy punktów: za egzamin (waga: 60%) weryfikujący wiedzę i umiejętności i za zaliczenia cząstkowe (waga: 40%).Szczegółowe warunki uzyskania oceny 3.5: Student uzyskał min. 60% sumy punktów: za egzamin (waga: 60%) weryfikujący wiedzę i umiejętności i za zaliczenia cząstkowe (waga: 40%).Szczegółowe warunki uzyskania oceny 4.0: Student uzyskał min. 70% sumy punktów: za egzamin (waga: 60%) weryfikujący wiedzę i umiejętności i za zaliczenia cząstkowe (waga: 40%).Szczegółowe warunki uzyskania oceny 4.5: Student uzyskał min. 80% sumy punktów: za egzamin (waga: 60%) weryfikujący wiedzę i umiejętności i za zaliczenia cząstkowe (waga: 40%).Szczegółowe warunki uzyskania oceny 5.0: Student uzyskał min. 90% sumy punktów: za egzamin (waga: 60%) weryfikujący wiedzę i umiejętności i za zaliczenia cząstkowe (waga: 40%). |

…………………………………………………………………………...

*(podpis kierownika jednostki prowadzącej zajęcia lub koordynatora przedmiotu)*