

KARTA MODUŁU

| I. OGÓLNE INFORMACJE O MODULE | | | | | | | |
|---|---|---|-----------|--------------|----------|-------------------------------------|---|
| COLLEGIUM WITELONA UCZELNIA PAŃSTOWA WYDZIAŁ NAUK TECHNICZNYCH I EKONOMICZNYCH | | | | | | | |
| Kierunek studiów: | INŻYNIERIA PRODUKCJI I LOGISTYKI | | | | | | |
| Poziom studiów: | studia drugiego stopnia | | | | | | |
| Profil studiów: | praktyczny | | | | | | |
| Forma studiów: | stacjonarne | | | | | | |
| Nazwa modułu: | Programowanie dla inżynierów | | | | | | |
| Rodzaj modułu: | Obowiązkowy | | | | | | |
| Język wykładowy: | Język polski* | | | | | | |
| Rok studiów: | 1 | Formy prowadzenia zajęć wraz z liczbą godzin dydaktycznych: | | | | | |
| Semestr: | 1 | Wykład | Ćwiczenia | Laboratorium | Warsztat | Projekt | Seminarium |
| Liczba punktów ECTS ogółem: | 3 | - | - | 15/10 | - | 15/8 | - |
| Forma zaliczenia: | Zaliczenie na ocenę | | | | | | |
| Wymagania wstępne: | Wiedza i umiejętności z informatyki na poziomie podstawowym szkoły ponadpodstawowej | | | | | | |
| II. CELE KSZTAŁCENIA | | | | | | | |
| Cele kształcenia: | | | | | | | |
| <p>Cel 1: Nabycie wiedzy i umiejętności w zakresie programowania w językach Python, C/ C++.</p> <p>Cel 2: Nauczenie się formułować podstawowe algorytmy na potrzeby przemysłowe wykorzystując między innymi języki programowania.</p> | | | | | | | |
| III. EFEKTY UCZENIA SIĘ WRAZ Z ODNIESIENIEM DO EFEKTÓW KIERUNKOWYCH ORAZ METODY WERYFIKACJI EFEKTÓW | | | | | | | |
| Efekt | Student, który zaliczył moduł w zakresie: | | | | | Odniesienie do efektów kierunkowych | Metody weryfikacji |
| wiedzy: | | | | | | | |
| W01 | Posiada wiedzę o językach programowania stosowanych w na potrzeby przemysłu. | | | | | K2IPL_W02 | Projekt programistyczny Zadania kontrolne na zajęciach |
| umiejętności: | | | | | | | |
| U01 | Student potrafi posługiwać się językiem programowania do rozwiązania wybranych problemów algorytmicznych, w tym dotyczące analizy danych przemysłowych. | | | | | K2IPL_U02 | Projekt programistyczny |
| U02 | Student potrafi przeanalizować oraz poprawić kod źródłowy programów napisanych w języku Python, C, C++. | | | | | K2IPL_U02 | Zadania kontrolne na zajęciach |
| kompetencji społecznych: | | | | | | | |
| - | | | | | | | |
| IV. TREŚCI PROGRAMOWE | | | | | | | |
| Treści programowe (tematyka zajęć, zaprezentowana z podziałem na poszczególne formy zajęć z określeniem liczby godzin potrzebnych na ich realizację) | | | | | | | |
| Laboratorium: | | | | | | | |
| Kod | Tematyka zajęć | | | | | Liczba godzin 15/10 | |

| | | |
|--|---|------------------------------------|
| lab1 | Wprowadzenie do algorytmiki. Zapoznanie się ze środowiskiem programowania, umiejętnością instalacji oraz konfiguracji środowiska. Pierwszy program. Proste algorytmy. | 4/2 |
| lab2 | Nauka programowania w zakresie instrukcji warunkowych oraz używania różnych typów danych | 2/2 |
| lab3 | Nauka programowania w zakresie instrukcji iteracyjnych oraz użycia tablicowego typu danych | 2/2 |
| lab4 | Programowanie z użyciem funkcji. Przekazywanie parametrów. | 2/1 |
| lab5 | Programowanie z interfejsem graficznym. | 2/1 |
| lab6 | Wykonanie zadań kontrolnych na zaliczenie. | 3/2 |
| Projekt: | | |
| Kod | Tematyka zajęć | Liczba godzin 15/8 |
| p1 | Określenie celu oraz założeń projektowych. Ustalenie wymagań, danych wejściowych oraz wynikowych. Omówienie algorytmów na potrzebę projektu. | 4/2 |
| p2 | Implementacja poszczególnych założeń projektowych w wybranym języku programowania. Testy funkcjonalne oraz akceptacyjne. | 8/4 |
| p3 | Sporządzenie dokumentacji projektowej. Zaliczenie projektu. | 3/2 |
| V. METODY KSZTAŁCENIA, NARZĘDZIA DYDAKTYCZNE | | |
| <p>1. Metody kształcenia: Zajęcia praktyczne z użyciem stanowiska komputerowego;</p> <p>2. Narzędzia (środki) dydaktyczne tablica multimedialna, prezentacje multimedialne, narzędzia nauczania na odległość, internet.</p> | | |
| VI. FORMA I KRYTERIA ZALICZENIA MODUŁU | | |
| <p>Forma zaliczenia modułu. Zaliczenie na ocenę</p> <p>Kryteria oceny formującej***: 1. Krótkie zadania domowe 2. Umiejętność samodzielnego rozwiązywania zadań przy stanowisku komputerowym 3. Sprawdzian praktyczny z programowania 4. Dokumentacja projektowa</p> <p>Kryteria oceny podsumowującej***: 1. Kolokwia pisemne 50-59% - ocena dostateczna, 60-69% - ocena dostateczna plus, 70-79% - ocena dobra, 80-89% - ocena dobra plus, powyżej 90% - ocena bardzo dobra</p> <p>Na ocenę 3,0: student zna podstawowe metody i narzędzia, potrafi przy pomocy prowadzącego rozwiązać proste zadania. Na ocenę 3,5: zna podstawowe metody i narzędzia, potrafi samodzielnie rozwiązać proste zadania. Na ocenę 4,0: zna metody i narzędzia omawiane na zajęciach, potrafi je samodzielnie zastosować. Z pomocą prowadzącego potrafi rozwiązać zadania typowe. Na ocenę 4,5: zna metody i narzędzia omawiane na zajęciach, potrafi je samodzielnie zastosować. Samodzielnie potrafi rozwiązać zadania typowe. Na ocenę 5: zna metody i narzędzia omawiane na zajęciach, potrafi je samodzielnie zastosować. Samodzielnie potrafi rozwiązać zadania typowe. Jest aktywny na zajęciach.</p> <p>Ocena podsumowująca***: Ocena z modułu: średnia ocen z poszczególnych form zajęć.</p> | | |
| VII. BILANS PUNKTÓW ECTS - NAKŁAD PRACY STUDENTA | | |
| Kategoria | | Obciążenie studenta S/N |
| Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela (godziny kontaktowe) | | 30/18 |
| Udział w wykładach | | - |

| | |
|--|--------------|
| Udział w innych formach zajęć (laboratorium, projekt**) | 30/18 |
| Inne: udział w egzaminie | - |
| Samodzielna praca studenta (godziny niekontaktowe) | 45/57 |
| Przygotowanie do wykładu | - |
| Przygotowanie do innych form zajęć (laboratorium, projekt**) | 25/32 |
| Przygotowanie do egzaminu | - |
| Przygotowanie do zaliczenia innych zajęć (laboratorium, projekt**) | 15/20 |
| Inne (np. gromadzenie materiałów do projektu, kwerenda internetowa, opracowanie prezentacji multimedialnej itp.) | 5 |
| Łączna liczba godzin | 75 |
| Punkty ECTS za moduł | 3 |

VIII. ZALECANA LITERATURA

Literatura podstawowa:

1. Michael Dawson, Python dla każdego. Podstawy programowania. Wydanie III , Helion, 2014
2. Brian W. Kernighan, Dennis M. Ritchie, Język ANSI C. Programowanie. Wydanie II, Helion, 2020

Literatura uzupełniająca:

1. Paul Barry, Python. Rusz głową! Wydanie II, Helion, 2017
2. Jerzy Grębosz, Opus magnum C++ 11. Programowanie w języku C++. Wydanie II poprawione, Helion, 2020

*należy odpowiednio wypełnić

**należy wpisać formę/formy przypisane do modułu określone w programie studiów (wykład, ćwiczenia, seminarium, konwersatorium, lektorat, laboratorium, warsztat, projekt, zajęcia praktyczne, zajęcia terenowe, zajęcia wychowania fizycznego, praktyka zawodowa, inne)

*** proszę wpisać odpowiednie kryteria oceny formującej i podsumowującej