



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Języki skryptowe [S1MiKC2>JS]

Przedmiot

Kierunek studiów

Mikroelektronika i komunikacja cyfrowa

Rok/Semestr

1/1

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

polski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

0

Laboratorium

30

Inne

0

Ćwiczenia

0

Projekty/seminaria

0

Liczba punktów ECTS

2,00

Koordynatorzy

dr inż. Paweł Sroka

pawel.sroka@put.poznan.pl

Wykładowcy

Wymagania wstępne

Student rozpoczynający naukę powinien posiadać podstawową wiedzę z matematyki i fizyki. Dodatkowo, powinien posiadać wiedzę i podstawowe umiejętności obsługi komputera typu PC. Student powinien rozumieć konieczność zdobywania nowej wiedzy i umiejętności związanych z realizowanym kierunkiem studiów.

Cel przedmiotu

Celem przedmiotu jest nabycie przez studentów podstawowej wiedzy i umiejętności z zakresu programowania z wykorzystaniem wybranych języków skryptowych. Dodatkowo, w ramach przedmiotu przekazana zostanie wiedza i umiejętności z zakresu trwałego zapisu i podstawowej wizualizacji danych przy użyciu wybranych języków skryptowych.

Przedmiotowe efekty uczenia się

Wiedza:

1. Ma podstawową wiedzę w zakresie zasad tworzenia programów komputerowych i wykorzystania poznanych języków skryptowych do implementacji wybranych algorytmów lub przetwarzania danych.

Umiejętności:

1. Potrafi opracować proste aplikacje korzystając z poznanych języków skryptowych w celu przeprowadzenia analizy i rozwiązywania problemów właściwych dla kierunku studiów.
2. Potrafi zapisać i przedstawić zebrane dane w formie liczbowej i graficznej korzystając z poznanych języków skryptowych.

Kompetencje społeczne:

1. Rozumie konieczność poszerzania wiedzy na temat korzystania z języków skryptowych; ma świadomość, że wiedza i umiejętności w tym obszarze szybko ewoluują.

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Umiejętności nabyte podczas laboratorium są weryfikowane podczas realizacji 8-15 ćwiczeń praktycznych polegających na implementacji oprogramowania o określonej funkcjonalności w wybranym języku skryptowym, zgodnie z przygotowaną instrukcją do zadania. Każde z ćwiczeń jest oceniane w postaci punktów na podstawie przesłanych rozwiązań, przy czym liczba punktów odpowiadająca zadaniu jest zależna od jego stopnia skomplikowania. Wpływ na ocenę końcową ma też ocena pracy i zaangażowania studenta na zajęciach, a także realizacja ewentualnych dodatkowych zadań domowych. Ocena końcowa zależna jest od liczby zdobytych punktów, przy czym próg zaliczeniowy wynosi 50% maksymalnej liczby punktów za wszystkie ćwiczenia. Progi dla kolejnych ocen kształtują się następująco:

- poniżej 50% - 2,0
- 50% do 60% - 3,0
- powyżej 60% do 70% - 3,5
- powyżej 70% do 80% - 4,0
- powyżej 80% do 90% - 4,5
- powyżej 90% - 5,0.

Treści programowe

W ramach zajęć omawiane są aspekty teoretyczne i praktyczne wykorzystania wybranych języków skryptowych do rozwiązywania wybranych problemów z zakresu matematyki, informatyki i telekomunikacji. Tematyka zajęć obejmuje podstawowe zagadnienia implementacji programów z wykorzystaniem języków skryptowych, w tym omówienie składni języków oraz sposobów zapisu i prezentacji danych.

Tematyka zajęć

W ramach laboratoriów realizowane są następujące tematy:

1. Podstawowe elementy i składnia wybranego języka skryptowego (np. Python): typy danych, operatory arytmetyczne i kolejność działań, operacje wejścia-wyjścia (4 godziny).
2. Działania iteracyjne (pętle) i podejmowanie decyzji (instrukcje warunkowe) w wybranym języku skryptowym (4 godziny).
3. Podstawowe struktury danych w wybranym języku skryptowym - tablice, listy, słowniki (tablice asocjacyjne) (6 godzin).
4. Programowanie proceduralne z wykorzystaniem wybranego języka skryptowego - funkcje i moduły. Funkcje wbudowane i tworzenie własnych funkcji (4 godziny).
5. Przetwarzanie danych z wykorzystaniem wybranych języków skryptowych - odczyt i zapis plików w różnych formatach (3 godziny).
6. Graficzna prezentacja danych (wyników) w wybranym języku skryptowym (3 godziny).
7. Detekcja i korygowanie błędów w wybranym języku skryptowym. Programowa obsługa błędów uruchomieniowych. Weryfikacja danych wejściowych (2 godziny).
8. Wykorzystanie wybranych bibliotek poznanego języka skryptowego (4 godziny).

Metody dydaktyczne

Ćwiczenia laboratoryjne: wykonanie zadań podanych przez prowadzącego z wykorzystaniem komputerów - ćwiczenia praktyczne, wspomagane prezentacją multimedialną.

Literatura

Podstawowa:

Mark Lutz, Python. Wprowadzenie. Wydanie V, Helion 2020

Piotr Wróblewski, "Algorytmy w Pythonie : techniki programowania dla praktyków", Helion, 2022

Uzupełniająca:

Robert Johansson, Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib, Helion 2021

Qingkai Kong, Timmy Siauw, Alexandre Bayen, "Python Programming and Numerical Methods - A Guide for Engineers and Scientists" (online)

www.w3schools.com (online)

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	60	2,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu)	30	1,00