

**WYDZIAŁ MATEMATYKI
KARTA PRZEDMIOTU**

Nazwa przedmiotu w języku polskim: **Programowanie**
 Nazwa przedmiotu w języku angielskim: **Computer Programming**
 Kierunek studiów: **Matematyka, Matematyka i Analiza Danych**
 Specjalność:
 Stopień studiów i forma: **I stopień, stacjonarna**
 Rodzaj przedmiotu: **wybieralny**
 Kod przedmiotu:
 Grupa kursów: **TAK**

| | Wykład | Ćwiczenia | Laboratorium | Projekt | Seminarium |
|---|-------------------------------------|-----------|--------------|---------|------------|
| Liczba godzin zajęć zorganizowanych w Uczelni (ZZU) | 30 | | 30 | | |
| Liczba godzin całkowitego nakładu pracy studenta (CNPS) | | | | | |
| Forma zaliczenia | Egzamin/ zaliczenie na ocenę* | | | | |
| Dla grupy kursów zaznaczyć kurs końcowy (X) | X | | | | |
| Liczba punktów ECTS | | | | | |
| w tym liczba punktów odpowiadająca zajęciom o charakterze praktycznym (P) | | | | | |
| w tym liczba punktów ECTS odpowiadająca zajęciom wymagającym bezpośredniego udziału nauczycieli lub innych osób prowadzących zajęcia (BU) | | | | | |

*niepotrzebne skreślić

WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I KOMPETENCJI SPOŁECZNYCH

1. Student zna dobrze podstawy wybranego języka programowania — *Wstęp do programowania.*
2. Student potrafi posługiwać się powłoką tekstową — *Wstęp do programowania.*
3. Student potrafi pracować w rozproszonym systemie kontroli wersji — *Wstęp do programowania.*

CELE PRZEDMIOTU

- C1 Zapoznanie się z podstawowymi zagadnieniami informatyki.
 C2 Pozyskanie umiejętności praktycznego zastosowania poznanej wiedzy, w szczególności implementacji prostych algorytmów.
 C3 Poznanie zasad projektowania aplikacji w paradygmacie programowania obiektowego.
 C4 Poznanie podstaw obsługi arkuszy kalkulacyjnych i tworzenia makr.

PRZEDMIOTOWE EFEKTY UCZENIA SIĘ

Z zakresu wiedzy:

PEU_W01 Student zna podstawowe pojęcia informatyki, takie jak program, algorytm i złożoność obliczeniowa.

PEU_W02 Student zna zasady projektowania aplikacji w paradygmacie programowania obiektowego.

PEU_W03 Student zna zasadę działania arkusza kalkulacyjnego i makr.

Z zakresu umiejętności:

PEU_U01 Student potrafi zaimplementować algorytm w wybranym języku programowania.

PEU_U02 Student potrafi zmierzyć czas wykonania programu i porównać go z wyznaczoną złożonością obliczeniową.

PEU_U03 Student potrafi zaimplementować program składający się z wielu klas.

PEU_U04 Student potrafi rozbić problem na logiczne bloki w formie funkcji, klas i modułów.

PEU_U05 Student potrafi obsługiwać arkusze kalkulacyjne i tworzyć makra.

Z zakresu kompetencji społecznych:

PEU_K01 Student jest przygotowany do zdobywania nowych kompetencji w zakresie informatyki i jej zastosowań w matematyce.

PEU_K02 Student jest przygotowany do pracy zespołowej nad projektami informatycznymi.

TREŚCI PROGRAMOWE

| Forma zajęć - wykład | | Liczba godzin |
|----------------------|--|---------------|
| Wy1 | Mierzenie czasu działania algorytmu; Złożoność obliczeniowa. | 2 |
| Wy2 | Przykładowe algorytmy wyszukiwania i poszukiwania. | 2 |
| Wy3 | Rekurencja i przykładowe algorytmy rekurencyjne. | 2 |
| Wy4 | Sprawdzanie pierwszości liczb; Rozkład na czynniki pierwsze. | 2 |
| Wy5 | Podstawowe algorytmy sortowania. | 2 |
| Wy6 | Algorytmy na napisach; Praca z czasem i datami. | 2 |
| Wy7 | Wstęp do programowania obiektowego; Atrybuty i metody; Metody specjalne; Iteratory i generatory. | 2 |
| Wy8 | Dziedziczenie w Pythonie; Abstrakcyjne klasy bazowe. | 2 |
| Wy9 | Metody klas i statyczne; Właściwości; Adnotacje typów; Diagramy UML. | 2 |
| Wy10 | Testy jednostkowe; Zasady dokumentowania kodu; Zasady projektowania aplikacji. | 2 |
| Wy11 | Przykład większej aplikacji w bibliotece PyGame. | 2 |
| Wy12 | Zaawansowane korzystanie z arkuszy kalkulacyjnych. | 2 |
| Wy13 | Podstawy języka VBA. | 2 |
| Wy14 | Tworzenie makr arkusza kalkulacyjnego w języku VBA. | 2 |
| Wy15 | Tworzenie makr arkusza kalkulacyjnego w języku Python. | 2 |

| | | |
|--|-------------|-----------|
| | Suma godzin | 30 |
|--|-------------|-----------|

| Forma zajęć - laboratorium | | Liczba godzin |
|-----------------------------------|---|----------------------|
| La1 | Przypomnienie składni języka Python; Ćwiczenia z mierzenia czasu wykonywania kodu. | 2 |
| La2 | Ćwiczenia dotyczące algorytmów wyszukiwania i poszukiwania. | 2 |
| La3 | Ćwiczenia dotyczące algorytmów rekurencyjnych. | 2 |
| La4 | Ćwiczenia dotyczące algorytmów wyznaczania i zliczania liczb pierwszych oraz ich zastosowania. | 2 |
| La5 | Ćwiczenia dotyczące algorytmów sortowania. | 2 |
| La6 | Ćwiczenia dotyczące algorytmów na napisach. | 2 |
| La7- La9 | Ćwiczenia dotyczące programowania obiektowego oraz przygotowywania diagramów UML. | 6 |
| La10 | Ćwiczenia dotyczące testów jednostkowych. | 2 |
| La11 | Ćwiczenia z pracy z istniejącym kodem – modyfikowanie przykładowej aplikacji udostępnionej podczas wykładu. | 2 |
| La12- La14 | Ćwiczenia z arkusza kalkulacyjnego oraz VBA. | 6 |
| La15 | Ćwiczenia z arkusza kalkulacyjnego oraz biblioteki xlwings. | 2 |
| | Suma godzin | 30 |

| STOSOWANE NARZĘDZIA DYDAKTYCZNE |
|--|
| N1. Wykład multimedialny z elementami tradycyjnego. N2. Laboratorium komputerowe. N3. Praca własna studenta. |

OCENA OSIĄGNIĘCIA PRZEDMIOTOWYCH EFEKTÓW UCZENIA SIĘ

| Oceny (F – formująca (w trakcie semestru), P – podsumowująca (na koniec semestru)) | Numer efektu uczenia się | Sposób oceny osiągnięcia efektu uczenia się |
|---|--|--|
| F1 | PEU_U01 PEU_U02 PEU_K01 | Listy zadań realizowane podczas laboratoriów. |
| F2 | PEU_W02 PEU_U01 PEU_U03 PEU_U04 PEU_K01 PEU_K02 | Projekt grupowy dotyczący zaprojektowania i stworzenia aplikacji wykorzystującej paradygmat programowania obiektowego. |
| F3 | PEU_W03 PEU_U04 PEU_U05 PEU_K01 | Sprawozdanie dotyczące wykorzystania makr w arkuszach kalkulacyjnych. |

| | | |
|---|---------|---|
| F4 | PEU_W01 | Kolokwium lub kartkówki podczas wykładu (modyfikator oceny, możliwe wartości: -1, -0.5, 0, +0.5) |
| $P = 0.3 \cdot F1 + 0.5 \cdot F2 + 0.2 \cdot F3 + F4$ | | |

LITERATURA PODSTAWOWA I UZUPEŁNIAJĄCA

LITERATURA PODSTAWOWA:

- [1] L. Ramalho, *Fluent Python*, O'Reilly 2022, wydanie 2
- [2] B. Slatkin, *Effective Python*, Addison-Wesley 2019, wydanie 2.
- [3] R. L. Graham, D. E. Knuth, O. Patashnik, *Matematyka konkretna*, Wydawnictwo naukowe PWN 2012, wydanie 4.
- [4] J. Green, S. Bullen, R. Bovey, M. Alexander, *Excel 2007 VBA Programmer's Reference*, Willey 2011, wydanie 1.

LITERATURA UZUPEŁNIAJĄCA:

- [1] D. Harrell, *Rzecz o istocie informatyki. Algorytmika*, WNT 2000, wydanie 1.
- [2] A. Hunt, D. Thomas. *Pragmatyczny programista. Od czeladnika do mistrza*. WNT 2002, wydanie 1.

OPIEKUN PRZEDMIOTU (IMIĘ, NAZWISKO, ADRES E-MAIL)

Andrzej Giniewicz (Andrzej.Giniewicz@pwr.edu.pl)
Wojciech Połowczuk (Wojciech.Polowczuk@pwr.edu.pl)
Szymon Żeberski, szymon.zeberski@pwr.edu.pl